10

TOUCHPAD®

Computer Applications

Teacher's Manual

Extended Support for Teachers

ORANGE®
www.orangeeducation.in
www.thetouchpad.com

Teacher's Time Table

Periods / Days	0	I	II	III	IV		V	VI	VII	VIII
Monday						B				
Tuesday						R				
Wednesday						E				
Thursday						A				
Friday						K				
Saturday										

Development Milestones in a Child

Development milestones are a set of functional skills or age-specific tasks that most children can do at a certain age. These milestones help the teacher to identify and understand how children differ in different age groups.

Age 5 - 8 Years	
Physical	• First permanent tooth erupts • Shows mature throwing and catching patterns • Writing is now smaller and more readable • Drawings are now more detailed, organised and have a sense of depth
Cognitive	• Attention continues to improve, becomes more selective and adaptable • Recall, scripted memory, and auto-biographical memory improves • Counts on and counts down, engaging in simple addition and subtraction • Thoughts are now more logical
Language	• Vocabulary reaches about 10,000 words • Vocabulary increases rapidly throughout middle childhood
Emotional/Social	• Ability to predict and interpret emotional reactions of others enhances • Relies more on language to express empathy • Self-conscious emotions of pride and guilt are governed by personal responsibility • Attends to facial and situational cues in interpreting another's feelings • Peer interaction is now more prosocial, and physical aggression declines

"If you cannot do great things, do small things in a great way."

Age 9 - 11 Years	
Physical	• Motor skills develop resulting enhanced reflexes
Cognitive	• Applies several memory strategies at once • Cognitive self-regulation is now improved
Language	• Ability to use complex grammatical constructions enhances • Conversational strategies are now more refined
Emotional/Social	• Self-esteem tends to rise • Peer groups emerge

Age 11 - 20 Years	
Physical	• If a girl, reaches peak of growth spurt • If a girl, motor performance gradually increases and then levels off • If a boy, reaches peak and then completes growth spurt • If a boy, motor performance increases dramatically
Cognitive	• Is now more self-conscious and self-focused • Becomes a better everyday planner and decision maker
Emotional/Social	• May show increased gender stereotyping of attitudes and behaviour • May have a conventional moral orientation

Managing the children's learning needs according to their developmental milestones is the key to a successful teaching-learning transaction in the classroom.

"Family is the most important thing in the world."

Teaching Pedagogies

Pedagogy is often described as the approach to teaching. It is the study of teaching methods including the aims of education and the ways in which such goals can be achieved.

Lesson Plans

A lesson plan is the instructor's road map which specifies what students needs to learn and how it can be done effectively during the class time. A lesson plan helps teachers in the classroom by providing a detailed outline to follow in each class.

A lesson plan addresses and integrates three key components:

- Learning objectives
- Learning activities
- Assessment to check the student's understanding

A lesson plan provides an outline of the teaching goals:

Before the class:

1. Identify the learning objectives.
2. Plan the lesson in an engaging and meaningful manner.
3. Plan to assess student's understanding.
4. Plan for a lesson closure.

During the class:

Present the lesson plan.

After the class:

Reflect on what worked well and why. If needed, revise the lesson plan.

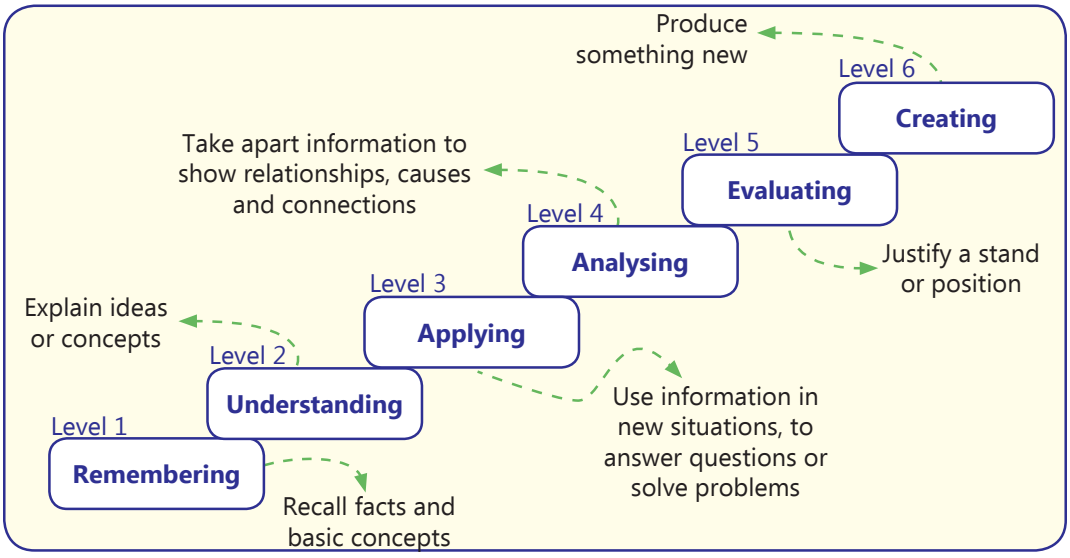
"Knowing yourself is the beginning of all wisdom."

Teaching Strategies

Numerous strategies have evolved over the years to facilitate the teaching-learning process in the classrooms.

Bloom's Taxonomy

Bloom's Taxonomy was created by **Dr Benjamin Bloom** and several of his colleagues, to promote higher forms of thinking in education instead of rote learning. There are three domains of learning: cognitive (mental), affective (emotional), and psychomotor (physical). However, when we refer to Bloom's Taxonomy we speak of the cognitive domain. Bloom's Taxonomy is a list of cognitive skills that is used by teachers to determine the level of thinking their students have achieved. As a teacher, one should attempt to move students up the taxonomy as they progress in their knowledge.



Teachers should focus on helping students to remember information before expecting them to understand it, helping them understand it before expecting them to apply it to a new situation, and so on.

"If you have no confidence in self, you are twice defeated in the race of life."

Class 10

LESSON PLAN

Touchpad Computer Applications

1. Introduction to Object-Oriented Programming Concepts

Teaching Objectives

Students will learn about

- ☞ Computer Language
- ☞ Difference between POP and OOP
- ☞ Programming Paradigms
- ☞ Principles of Object-Oriented Programming

Number of Periods	
Theory	Practical
3	2

Teaching Plan

Before starting the chapter, make the students to revise about programming languages taught in earlier class and chapters for better understanding of the current topic.

Tell the students about the types of computer languages in detail which are:

Low-Level Language

High-Level Language

Also, explain the advantages and disadvantages of these in detail.

Share a comparison between three types of computer languages in detail with the students.

Define the programming paradigms to the students. Explain the procedure-oriented programming along with the principles, advantages and disadvantages of the same. Share the difference between POP and OOP in detail with the students.

POP	OOP
It divides the programs into small parts known as methods.	It divides the program into objects.
It follows top-down approach.	It follows bottom-up approach.
It deals with algorithm.	It deals with data.
It is less secure.	It is more secure.
Examples: C, Fortran	Examples: C++, Java

Tell the students about the principles of OOPS along with its four principles:

- Encapsulation
- Data Abstraction
- Inheritance
- Polymorphism

Tell the students about the difference between Abstraction and Encapsulation.

Abstraction	Encapsulation
Issues at the design level can be solved.	Issues at the implementation level can be solved.
Abstraction is about hiding unwanted details and showing only the essential information.	In encapsulation the methods and data are kept in a single unit.
Abstraction focuses on what the information object must contain.	It hides the internal details of how an object does something for security reasons.

Extension

Ask the students some oral questions based on this chapter.

Q. Define computer languages.

Q. Write the difference between LLL and HLL.

Q. What is OOP?

Q. What is POP?

Q. Write the difference between OOP and POP.

Q. Define the following:

- Encapsulation
- Data Abstraction
- Inheritance
- Polymorphism

Q. Write the difference between Abstraction and Encapsulation.

Evaluation

After explaining the chapter, let the students do the exercises given on Pages 18 to 20 in the main course book as **Mind Drill and Previous Years' Questions**.

2. Elementary Concept of Objects and Class

Teaching Objectives

Students will learn about

- ☞ Class in Java
- ☞ Object
- ☞ Properties of Class and Object

Number of Periods	
Theory	Practical
3	2

Teaching Plan

Before starting the chapter, make the students to revise about JAVA topics taught in earlier chapter for better understanding of the current topic.

Explain the concept of Classes and Objects in detail to the students with proper examples for better understanding of the topic.

Define Class in java to the students in detail for better understanding along with the components:

- Access Specifier
- "class" keyword
- Class name
- Data members
- Methods

Tell the students how to define a class in JAVA with proper examples and steps.

```
class Square{
//Data Members
int side;          // used for storing the side of the square
//Member Methods
void accept()      // Accepts the side from the user
void area()        // Calculates the area of the square
void perimeter()// Calculates the perimeter of the square
}
```

Define Objects in JAVA to the students along with the steps to create an object of a class. Also, tell them how to pass message between objects.

Share the difference between Class and Object:

Abstraction	Encapsulation
Class is the blue print from which objects are created	Object is an instance of a class.
It is a logical entity.	It is a physical entity.
No memory space is acquired when it is created.	Memory space is allocated when an object is created
Used to define only once.	It is created as many times as it is needed.

Define the properties of Class and Object to the students:

- Class is an Object Factory
- Object is an Instance of a Class
- Class is a User-defined Data Type

Extension

Ask the students some oral questions based on this chapter.

Q. Define Class.

Q. Define Object.

Q. What is the difference between a class and an object?

Q. Define the following components of a class:

- Access Specifier
- "class" keyword
- Class name
- Data members
- Methods

Q. Define the properties of an object.

Evaluation

After explaining the chapter, let the students do the exercises given on Pages 24 to 26 in the main course book as Mind Drill and Previous Years' Questions.

3. Values and Types

Teaching Objectives

Students will learn about

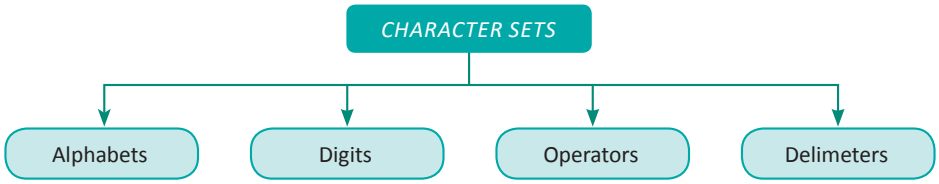
- ☞ Character Sets in Java
- ☞ Escape Sequences
- ☞ Data Types
- ☞ Declaration Vs Initialization
- ☞ Properties of Class and Object
- ☞ How Characters are Stored in Memory?
- ☞ Token
- ☞ Variable
- ☞ Type Conversion

Number of Periods	
Theory	Practical
3	2

Teaching Plan

Before starting the chapter, make the students to revise about JAVA topics taught in earlier chapter for better understanding of the current topic.

Define the character sets in Java to the students:



Tell the students about how characters are stored in memory. Also, tell them the types of encoding procedures along with advantages and disadvantages of the same:

- ASCII Code
- Unicode

Share with the students about the difference between ASCII and Unicode:

ASCII Code	Unicode
ASCII stands for American Standard Code for Information Interchange.	Unicode stands for Universal Character Encoding.
Standard ASCII character set only supports 128 characters (character range whose numeric value is between 0–127).	Unicode supports a wide range of characters. (0–221).
ASCII only uses one byte to represent each character.	Unicode supports up to 4 bytes for each character.
Unicode requires more space.	ASCII code requires less space.

Define the meaning of Escape Sequences and describe each of them with their purpose:

Escape Sequence	Description
\t	Used to insert a Horizontal tab.
\b	Used to insert a backspace.
\n	Used to insert a newline.
\r	Used to insert a carriage return.
\f	Used to insert a form feed.
\'	Used to insert a single quote character.
\"	Used to insert a double quote character.
\\	Used to insert a backslash character.

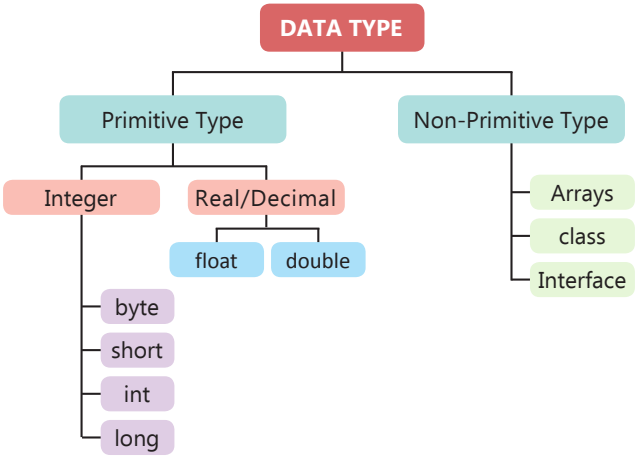
Tell the students about how to use Escape Sequence in BlueJ and tell their purpose with an example program:

- Horizontal Tab
- New Line
- Double Quote
- Single Quote
- Backspace
- Form Feed
- Carriage Return
- Back Slash

Explain the meaning of token to the students along with the different types of tokens which are:

- Keywords
- Identifiers
- Literals
- Operators
- Punctuators
- Separators

Share the meaning of data types with the students along with the types:



Define Variable to the students along with the functions you can perform on it:

- Declaring a variable
- Initialize a variable

Define the Declaration Vs Initialization to the students in detail. Also, define the types of initialization which are:

- Static Initialization
- Dynamic Initialization

Define the meaning and purpose of type conversion which are:

- Implicit Type
- Explicit Type

Extension

Ask the students some oral questions based on this chapter.

Q. What are character set?

Q. Define:

- Alphabets
- Digits
- Operators
- Delimeters

Q. What is the difference between ASCII Code and Unicode?

Q. What are escape sequence?

Q. Define the function of the following:

- Horizontal Tab
- New Line
- Double Quote
- Single Quote
- Backspace
- Form Feed
- Carriage Return
- Back Slash

Q. What is a token?

Q. Define the following:

- Keywords
- Identifiers
- Literals
- Operators
- Punctuators
- Separators

Q. What is data type?

Q. What is Declaration?

Q. What is initialization?

Q. Define the types of initialization:

a. Static

b. Dynamic

Q. What is data type conversion?

Evaluation

After explaining the chapter, let the students do the exercises given on Pages 41 to 47 in the main course book as Mind Drill and Previous Years' Questions.

4. Operators in JAVA

Teaching Objectives

Students will learn about

- ☞ Operators and Operands
- ☞ Types of Operators
- ☞ Forms of Operators
- ☞ Hierarchy of Operators

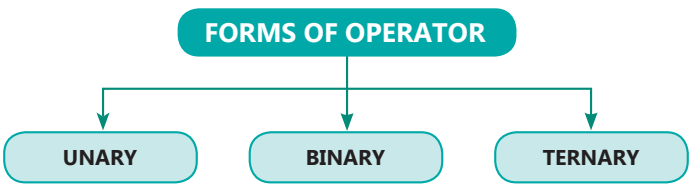
Number of Periods	
Theory	Practical
3	1

Teaching Plan

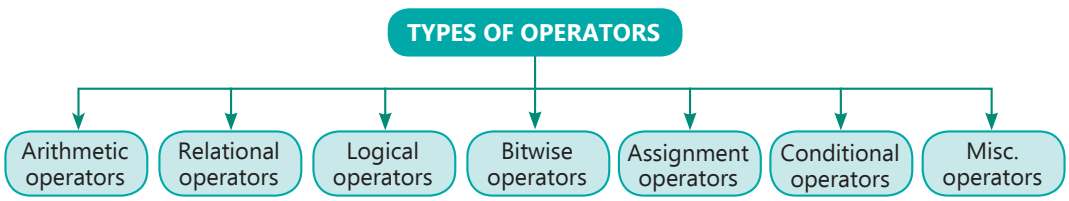
Before starting the chapter, make the students to revise about JAVA topics taught in earlier chapters for better understanding of the current topic.

Tell the students what is an operator and how it is used in Java in detail with suitable examples.

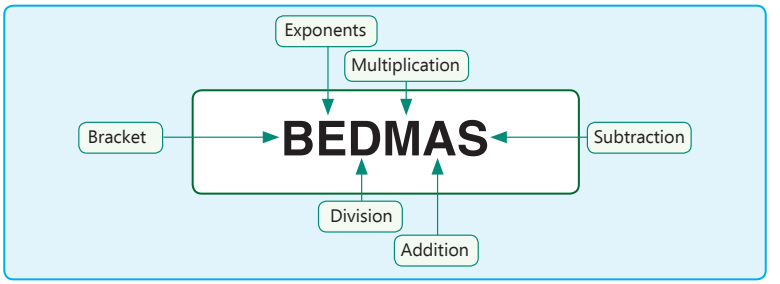
Define the following forms of operators which are divided into three parts:



Share the types of operators with the students in detail:



Explain the hierarchy of operators to the students for the better understanding of the topic in detail:



Hierarchy	Operators	Precedence
1.	postfix unary	++, --
2.	unary including prefix, logical NOT	++, --, +, -, ~ ,!
3.	multiplication, division and modulus	*, /, %
4.	addition and subtraction	+, -
5.	shift	<<, >>, >>>
6.	relational	<, >, <=, >=
7.	equality, non-equality	==, !=
8.	bitwise AND	&
9.	bitwise exclusive OR	^
10.	bitwise inclusive OR	|
11.	logical AND	&&
12.	logical OR	||
13.	ternary	? :
14.	assignment including shorthand	=, +=, -=, *=, /=, %=, &=, ^=, |=, <<=, >>=, >>>=

Tell the students about the associativity of operators which is:

Operators	Associativity
postfix unary	Right to Left
unary including prefix, logical NOT	Right to Left
multiplication, division and modulus	Left to Right
addition and subtraction	Left to Right

Define the purpose of Output statement to the students in detail.

Extension

Ask the students some oral questions based on this chapter.

Q. What is an operator?

Q. Define the forms of operators.

Q. What are the types of operators?

Q. Explain the hierarchy of operators.

Q. What is associativity of operators?

Evaluation

After explaining the chapter, let the students do the exercises given on Pages 64 to 70 in the main course book as **Mind Drill and Previous Years' Questions**.

5. Input in JAVA

Teaching Objectives

Students will learn about

- ☞ Introducing Packages
- ☞ Using Parameters
- ☞ Using InputStreamReader Class
- ☞ Exception
- ☞ Initialization
- ☞ Using Scanner Class
- ☞ Errors in Java
- ☞ Comments in Java

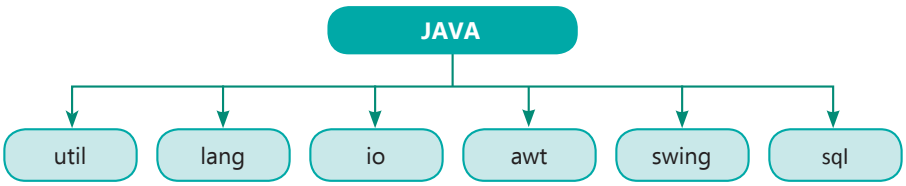
Number of Periods	
Theory	Practical
4	2

Teaching Plan

Before starting the chapter, make the students to revise about JAVA topics taught in earlier chapters for better understanding of the current topic.

Tell the students what are packages in Java and their types in detail:

Also tell them the types of comments:



Explain the initialization to the students along with the ways:

1. The value is assigned at the time of declaration.
2. The value is assigned after declaring the variable.

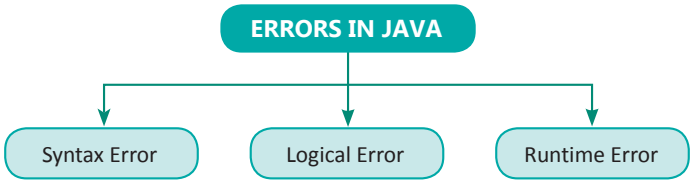
Tell the students about how to use input using parameters by defining two ways:

- Using Command Line Arguments
- During Method Call

Define the use of Scanner class to the students with the help of suitable programs, methods and examples.

Tell the students how to use InputStreamReader Class along with the methods used for the same.

Demonstrate what are errors in Java along with the types, examples and programs:



Define the Exceptions to the students in detail along with the types of the same:

- Checked Exception
- Unchecked Exception

Tell the students about the Exception Handling and The Try-catch Statement in detail with their purpose.

Tell the students that while writing a program code, we might need to explain certain parts of the program. This helps us to understand the reason to write the statements. This can be done with the help of **"Comments"**.

Also tell them the types of comments:

- Single line
- Multiline
- Documentation

Extension

Ask the students some oral questions based on this chapter.

Q. What are comments?

Q. What are packages?

Q. What is initialization?

Q. How to use Input in JAVA?

Q. What is the use of Scanner class?

Q. What are errors in Java?

Q. Define the following:

a. Exception Handling

b. Try-catch Statement

Evaluation

After explaining the chapter, let the students do the exercises given on Pages 85 to 89 in the main course book as **Mind Drill and Previous Years' Questions**.

6. Mathematical Library Methods

Teaching Objectives

Students will learn about

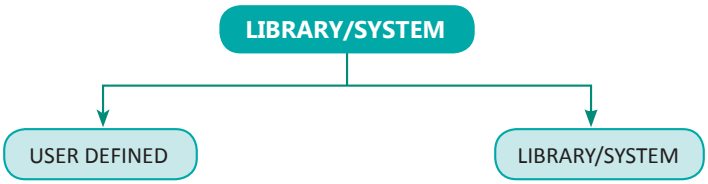
☞ Different Mathematical Methods

Number of Periods	
Theory	Practical
3	1

Teaching Plan

Before starting the chapter, make the students to revise about JAVA topics taught in earlier chapters for better understanding of the current topic.

Tell the students that there are two types of methods in Java:



Define Java.lang package to the students and tell them it is a default package.

Define the mathematical methods to the students along with the function, syntax and description:

Function	Description	Return Data Type	Syntax
min(a,b)	Returns the smaller number between a and b	int/long/float/double	Math.min(a,b);
max(a,b)	Returns the greater number between a and b	int/long/float/double	Math.max(a,b);
sqrt(a)	Returns the square root of a positive number.	double	Math sqrt(a):
cbit(a)	Returns the cube root of a number.	double	Math.cbrt(a);
pow(a,b)	Returns the value ab.	double	Math.pow(a,b);
abs(a)	Returns the absolute value (magnitude) of any number.	int/long/float/double	Math.abs(a):

round(a)	Returns the rounded value up to the nearest integer.	int/long	Math.round(a):
floor(a)	Returns the rounded value down to the nearest integer.	double	Math.floor(a);
ceil(a)	Returns the whole number greater than or equal to the number.	double	Math.ceil(a);
rint(a)	Returns the truncated value number.	double	Math.rint(a);
random()	Returns a random number between 0 and 1.	double	Math.random();
log(a)	Returns the value of natural logarithm of a number.	double	Math.log(a):
exp(a)	Returns an exponent value, i.e., e	double	Math.exp(a);
sin(a) cos(a) tan(a)	Returns the Sine/Cosine/Tangent of an angle a (expressed in radians).	double	Math.sin(a); Math.cos(a): Math.tan(a):

Extension

Ask the students some oral questions based on this chapter.

Q. What are methods in Java?

Q. Define two types of methods.

Q. Define mathematical methods.

Evaluation

After explaining the chapter, let the students do the exercises given on Pages 98 to 103 in the main course book as **Mind Drill and Previous Years' Questions.**

7. Conditional Construct in Java

Teaching Objectives

Students will learn about

- ☞ Flow of Control
- ☞ Ternary Operator
- ☞ Fall-Through Situation
- ☞ Difference Between if Statement and switch Statement
- ☞ Menu Driven Program using switch Statement
- ☞ Terminate Program Using System.exit()
- ☞ Conditional Statements
- ☞ Switch Case Statement

Number of Periods	
Theory	Practical
3	2

Teaching Plan

Before starting the chapter, make the students to revise about JAVA topics taught in earlier chapters for better understanding of the current topic.

Tell the students about the flow of control and define the types also:

- Normal
- Conditional
- Multiple Branching

Explain the conditional statements along with types, syntax and examples:

- if statement
- if...else statement
- if and only if statement
- if...else...if statement
- nested if statement

Tell the students about the ternary operators along with examples and programs.

Define the purpose of Switch statement along with some programs for better understanding to the students.

Demonstrate the Fall through situation to the students with the help of suitable examples and programs.

Explain the difference between IF statement and Switch statement to the student:

if statement	switch statement
The flow of control is bidirectional.	The flow of control is multidirectional i.e. depending on the choice.
All kinds of the relational operators are used for checking.	Checking is satisfied if the choice variable is matching the case value.
Any type of data types can be used for checking.	Only int and char data types are used for checking.
Satisfied condition returns true else returns false.	Satisfied condition neither returns true nor false.

Explain how to terminate a program to a student using System.exit( ) with the help of proper programs and examples.

Extension

Ask the students some oral questions based on this chapter.

Q. What is flow of control?

Q. Define the following types of flow of control:

a. Normal

b. Conditional

c. Multiple

Q. What is the difference between IF statement and Switch statement?

Evaluation

After explaining the chapter, let the students do the exercises given on Pages 126 to 136 in the main course book as **Mind Drill and Previous Years' Questions.**

8. Iterative Constructs in Java

Teaching Objectives

Students will learn about

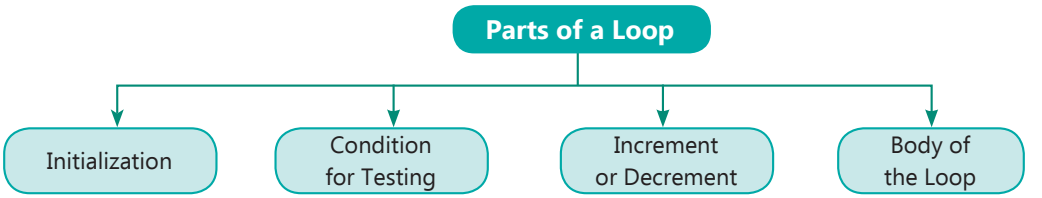
- ☞ Different Parts of a Loop
- ☞ Types of Loops
- ☞ The while Loop
- ☞ Different Forms of Loops
- ☞ Interconversion Between Different Types of Loops
- ☞ Jump Statements
- ☞ Categories of Loops
- ☞ The for Loop
- ☞ The do-while Loop

Number of Periods	
Theory	Practical
3	2

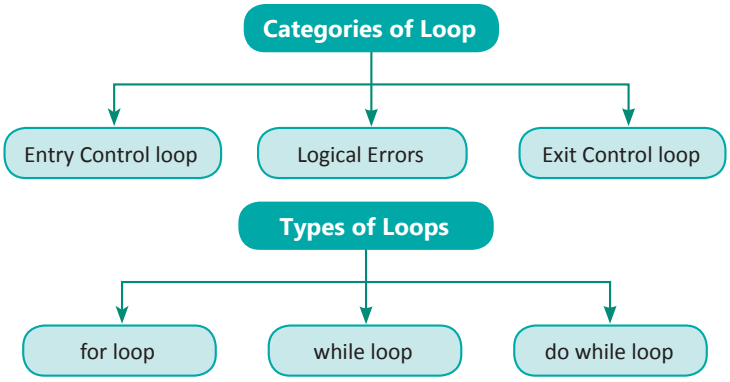
Teaching Plan

Before starting the chapter, make the students to revise about JAVA topics taught in earlier chapters for better understanding of the current topic.

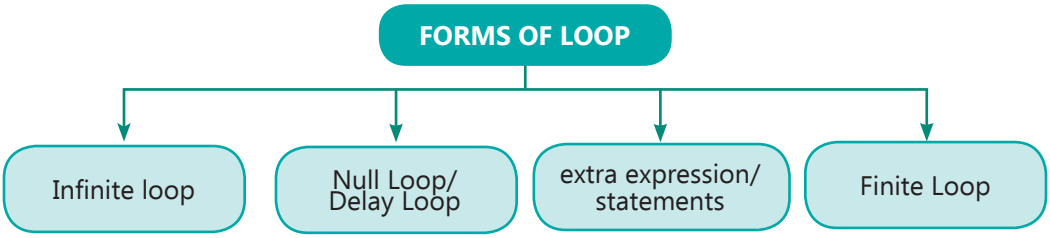
Define the meaning of Loops to the students and different parts of a loop:



Explain the categories and types of loop in detail to the students:



Define the different forms of loops to the students in detail with proper examples and programs:



Define the interconversion between different types of Loop to the students.

```
for (initialization; condition for testing; increment or decrement)
{
     // job performed by the body of the loop;
}
```

```
initialization;
while (condition for testing)
       {
          // job performed by the body of the loop;
          increment or decrement;
       }
```

```
initialization;
do
{
    // job performed by the body of the loop;
    increment or decrement;
} while (condition for testing);
```

Explain how to use continue statement with the help of proper examples and programs for better understanding.

Define the return statement with the help of proper examples and programs for better & clear understanding.

Extension

Ask the students some oral questions based on this chapter.

Q. What is loop?

Q. Define different parts of a loop.

Q. Write about categories of loop.

Q. What are the types of loop?

Q. What are the forms of loop?

Evaluation

After explaining the chapter, let the students do the exercises given on Pages 152 to 178 in the main course book as **Mind Drill and Previous Years' Questions**.

9. Nested Loop

Teaching Objectives

Students will learn about

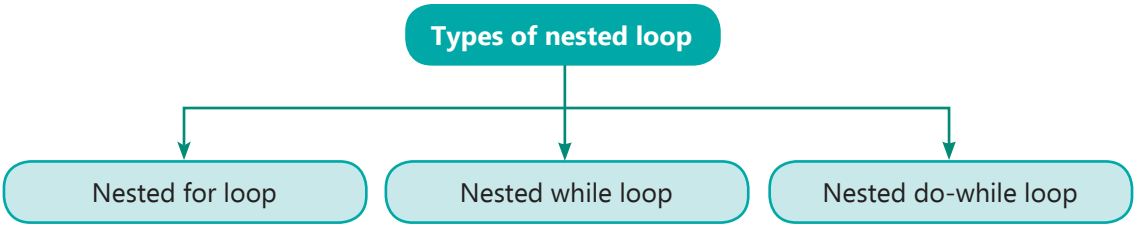
- ☞ Nested for Loop
- ☞ Nested while Loop
- ☞ Nested do-while Loop
- ☞ Using the break Statement in Nested Loops
- ☞ Using the continue Statement in Nested Loop
- ☞ Pattern Programs using Nested Loop

Number of Periods	
Theory	Practical
3	2

Teaching Plan

Before starting the chapter, make the students to revise about JAVA topics taught in earlier chapters for better understanding of the current topic.

Tell the students about Nested loop and share their types also.



Define nested for loop in detail with suitable example and programs.

```
for (initialization; condition; increment) // step of outer loop
{
for (initialization; condition; increment) // step of inner loop
{
// statement of inside loop
}
// statement of outer loop
}
```

Tell the students about the Nested While loop.

```
while(condition) //condition of outer loop
{
initialization; //initialization of inner loop
while(condition) //condition of inner loop
{
// statement of inside loop
increment/decrement; //incre or decre of inner loop
}
// statement of outer loop
increment/decrement; //incre or decre of outer loop
}
```

Tell the students about nested Do-while loop.

```
initialization; //initialization of outer loop
do { //outer do loop
initialization; //initialization of inner loop
```

```
do { //outer do loop
// statement of inside loop
increment/decrement; //incre or decre of inner loop
} while(condition); //condition of inner loop
// statement of outer loop
increment/decrement; //incre or decre of outer loop
} while(condition); //condition of outer loop
```

Explain how to use break statement with the help of proper examples and programs for better understanding.

Explain how to use continue statement with the help of proper examples and programs for better understanding.

Extension

Ask the students some oral questions based on this chapter.

Q. What is nested loop?

Q. What is nested for loop?

Q. What is nested while loop?

Q. What is nested do-while loop?

Q. Write the use of break statement.

Q. Write the use of continue statement.

Evaluation

After explaining the chapter, let the students do the exercises given on Pages 197 to 202 in the main course book as **Mind Drill and Previous Years' Questions.**

10. Class as the Basis of All Computation

Teaching Objectives

Students will learn about

- ☞ Object Class
- ☞ Creating Objects of a Class
- ☞ Different Component of a Class
- ☞ Nested Class
- ☞ Class as an Object Factory
- ☞ Using this Keyword
- ☞ Difference between Class and Object

Number of Periods	
Theory	Practical
3	2

Teaching Plan

Before starting the chapter, make the students to revise about JAVA topics taught in earlier chapters for better understanding of the current topic.

Define the meaning of an Object in Java to the students along with the:

- Characteristics or attributes
- Behaviour or methods

Tell the students about the Class and give suitable examples for better understanding.

Show the difference between Class and Object to the students.

Class	Object
A class is a logical entity.	An object is a real-world entity.
A class is a blueprint to create several objects.	An object is the instance of a class.
A class doesn't take any space in memory.	An object takes space in memory.

Tell the students about how to create objects of a class along with the detailed steps for defining a class.

Explain the different components of a class to the students which are:

- Access Specifiers
- Data Members
  - i. Instance variables
  - ii. Class variables
  - iii. Local variables
- Static Member Methods
- Constructors
- Member methods

Tell the students about the nested class and how to declare it with the help of a suitable example.

Show how class act as an object factory to the students.

Extension

Ask the students some oral questions based on this chapter.

Q. What is an object?

Q. What is a class?

Q. Write the difference between a class and an object.

Q. How to create object of a class?

Q. Define the different components of a class.

Q. What is nested class?

Q. Define class as an object factory.

Evaluation

After explaining the chapter, let the students do the exercises given on Page 225 to 239 in the main course book as Mind Drill and Previous Years' Questions.

11. User-Define Methods

Teaching Objectives

Students will learn about

- ☞ Need for Methods
- ☞ Defining a Method
- ☞ Actual and Formal Parameters
- ☞ The static and non-static Methods
- ☞ Method Overloading
- ☞ Forms of Methods
- ☞ Ways to Define a Method
- ☞ Invoking a Method
- ☞ Pure and Impure Methods

Number of Periods	
Theory	Practical
3	2

Teaching Plan

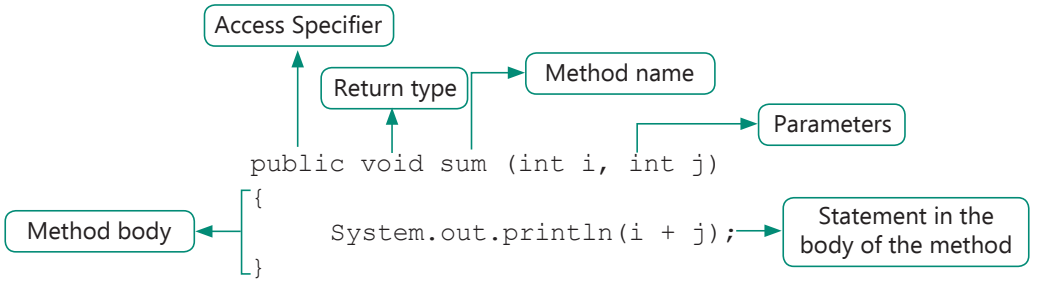
Before starting the chapter, make the students to revise about JAVA topics taught in earlier chapters for better understanding of the current topic.

Explain the need for Methods to the students in detail.

Tell the forms of Methods to the students in detail:

- Predefined Methods
- User Defined Methods

Explain how to define a Method to the students in detail with a suitable example.



Define the different components of a Method in detail to the students which are:

- Header
- Method Signature
- Access Specifier
- Return Type and Return Statement
- Method Name
- Parameters
- Body of the Method

Define the two ways to define a Method to the students with suitable examples and programs.

Explain the actual and formal parameters in detail to the students with suitable examples and programs.

Show the students how to invoke a Method with the help of examples and program related to it.

Explain the difference between Pass by Value and Pass by Reference:

Pass by Value	Pass by Reference
The copy of the actual parameters is sent to the formal parameter.	The actual parameters and the formal parameters share the same location in the memory.
Any change in the formal parameter will not reflect on the actual parameter.	Any change in the formal parameter while the method is executing, the actual parameter is also changed.
Only primitive data can be used.	Only non-primitive data can be used.

Explain the meaning of static and non-static methods in detail to the students for better understanding.

Demonstrate the Pure and Impure Methods to the students with the help of programs for clear understanding of topics to the students.

Define the method overloading to the students along with the advantages, disadvantages and types of the same.

Extension

Ask the students some oral questions based on this chapter.

Q. What is the need for methods?

Q. What are the forms of methods?

Q. How to define a method?

Q. Explain the different components of a method.

Q. Explain the ways to define a method.

Q. Define the following:

a. Actual Parameters

b. Formal Parameters

c. Pass by Value

d. Pass by Reference

Q. How to invoke a method?

Q . What is method overloading?

Evaluation

After explaining the chapter, let the students do the exercises given on Page 261 to 272 in the main course book as **Mind Drill and Previous Years' Questions**.

12. Constructors

Teaching Objectives

Students will learn about

- ☞ Introducing Constructor
- ☞ Different Types of Constructors
- ☞ Constructor Overloading
- ☞ Difference between Constructor and Method
- ☞ Similarity between Constructor and Method

Number of Periods	
Theory	Practical
3	2

Teaching Plan

Before starting the chapter, make the students to revise about JAVA topics taught in earlier chapters for better understanding of the current topic.

Introduce constructors to the student with the help of syntax, examples and programs associated with it.

Tell the students about how to call a constructor with the example and programs.

Explain the characteristics of a constructor to the students with the help of examples and programs.

Tell the different types of constructors to the students with the help of examples and programs. Different types of constructors are:

- Default
- Parameterised
- Non-Parameterised
- Copy

Explain the constructor overloading with the help of examples and programs for better understanding.

Define the difference between constructor and method to the students with help of the following table:

Constructor	Method
It is used to initialize the instance variables of the class.	It is used to execute the Java code which performs a specific job.
It can be invoked only once at the time of creating the object.	It can be invoked anytime as and when required.
It has the same name as that of the class.	It has the different name from the class.
It does not have a return type.	It has a return type.
It always public.	It may be public or private.

Explain the similarity between constructor and method with the help of example in some programs to define it and for better understanding.

Extension

Ask the students some oral questions based on this chapter.

Q. What is Constructor?

Q. Explain different types of constructors.

Q. Define the following:

- Default Constructors
- Parameterised Constructors
- Non-Parameterised Constructors
- Copy Constructors

Q. What is constructor Overloading?

Q. Write the difference between constructor and method.

Q. What are the similarities between a constructor and a method?

Evaluation

After explaining the chapter, let the students do the exercises given on Page 293 to 300 in the main course book as **Mind Drill and Previous Years' Questions.**

13. Library Classes

Teaching Objectives

Students will learn about

- ☞ Primitive and Composite Data Types
- ☞ Class as a Composite Data Type

- ☞ Difference Between Primitive and Composite Data Types
- ☞ Introduction to Wrapper Classes
- ☞ How to use Wrapper Classes?
- ☞ Autoboxing and Unboxing

Number of Periods	
Theory	Practical
3	2

Teaching Plan

Before starting the chapter, make the students to revise about JAVA topics taught in earlier chapters for better understanding of the current topic.

Define the following types of data to the students with the help of suitable examples and program:

- Primitive Data Type
- Composite Data Type

Define Class as a Composite Data Type to the students with the help of examples.

Explain the difference between primitive and composite data types:

Primitive Data Type	Composite Data Type
It predefined data type.	It is user-defined data type.
It always contains a value.	It can also contain null values.
It always starts with lowercase letter.	It can start with lowercase or uppercase letter.

Introduce the students with wrapper classes and tell the students that there are eight wrapper classes in java.lang package:

Wrapper Class	Primitive Data Type
Boolean	boolean
Character	char
Byte	byte
Short	short
Integer	int
Long	long
Float	float
Double	double

Demonstrate the need of the wrapper classes along with the use of wrapper classes to the students in detail by showing proper ways to do so.

Explain the meaning of autoboxing and unboxing to the students along with the following:

Parse Functions

Function	Format	Description
Integer.parseInt(String)	int n-Integer.parseInt("967");	It converts a string into integer form and stores it in integer variable.
Integer.valueOf(String)	int n=Integer.valueOf("2876");	
Long parseLong(String)	long a=Long.parseLong("875587");	It converts a string into long type data and stores it in a long variable
Long.valueOf(String)	long b=Long.valueOf("658765");	
Float.parseFloat(String)	float g=Float.parseFloat("89.32");	It converts a string into float type data and stores it in a float variable.
Float.valueOf(String)	float h=Float.valueOf("79.35");	
Double.parseDouble(String)	double a; a = Double.parse Double("65.125894");	It converts a string into double type data and stores it in a double variable.
Double.valueOf(String)	double b; b=Double.valueOf("69.1248753");	

Character Functions

Function	Format	Description
Character.isLetter()	char n='m'; Character.isLetter(n)	It checks whether a character 'n' is a letter or not and returns true for the given character.
Character.isDigit()	char a='9'; Character.isDigit(a) where 'a' is a character type variable containing a digit.	It checks whether a character is a digit or not and returns true for the given character.

Character.isLetterOr-Digit()	boolean b; b=Character.isLetterOrDigit('Q');	It checks if b stores a character. It returns true if character Q is a letter.
	boolean b; b=Character.isLetterOrDigit('5');	It checks if b stores a digit. It returns true if '5' is a digit.
Character.is WhiteSpace()	boolean b; b=Character.is WhiteSpace(' ');	It checks if b stores a whitespace. It returns true if it stores a white space.
	boolean b; b=Character.is WhiteSpace('*');	In this case, it will return false because * is not whitespace.
Character.is Upper-Case()	Character.is UpperCase(f) where f is a character type variable, i.e., char f='D';	It checks whether a character is in upper case or not. In this case it returns True.
Character.isLower Case()	Character.isLowerCase(g) where g is a character type variable	It checks whether a character is in lower case or not.
Character.toUpper-Case()	l = Character.toUpperCase(q) where q is a character type variable	It returns an uppercase character.
Character.toLower-Case()	l = Character.toLowerCase(q) where p is a character type variable	It returns a lowercase character.

Extension

Ask the students some oral questions based on this chapter.

Q. What is Primitive Data Type?

Q. What is Composite Data Type?

Q. Write the difference between primitive and composite data type.

Q. Define Class as a Composite Data Type.

Q. What are wrapper classes?

Q. What are parse functions?

Q. What are character functions?

Evaluation

After explaining the chapter, let the students do the exercises given on Page 311 to 315 in the main course book as **Mind Drill and Previous Years' Questions**.

14. Encapsulation and Inheritance

Teaching Objectives

Students will learn about

- ☞ Encapsulation
- ☞ Scope of Variables
- ☞ Access Specifiers
- ☞ Inheritance

Number of Periods	
Theory	Practical
3	2

Teaching Plan

Before starting the chapter, make the students to revise about JAVA topics taught in earlier chapters for better understanding of the current topic.

Define the meaning of encapsulation to the students in detail along with the related factors for defining its role.

```
class encapsulation_example
    {
        int a;
        String n;
        void input (int a1, String n1)
        {
            a=a1;
            n=n1;
        }
        void display ()
        {
            System.out.println (a);
            System.out.println (n);
        }
    }
```

Class

Data Members

Member Methods

Define Access Specifiers along with suitable examples and programs. Also tell the students about the three types of access specifiers in Java:

- Private
- Public
- Protected

Define the scope of variables with the help of programs to the students. Also, define the types of variables:

- Local
- Class
- Instance
- Argument

Tell the students about inheritance and different type of inheritance in detail with examples:

- Single Inheritance
- Multiple Inheritance
- Multilevel Inheritance
- Hierarchical Inheritance
- Hybrid Inheritance

Explain why inheritance is required to the students in detail.

Extension

Ask the students some oral questions based on this chapter.

Q. What is encapsulation?

Q. What are access specifiers?

Q. Define the types of access specifiers.

Q. What are scope of variables?

Q. What is inheritance?

Q. Explain the types of inheritance.

Q. Why inheritance is required?

Evaluation

After explaining the chapter, let the students do the exercises given on Pages 326 to 329 in the main course book as **Mind Drill and Previous Years' Questions**.

15. Arrays

Teaching Objectives

Students will learn about

- What is an Array?
- Types of Arrays
- Length of an Array
- Difference between Single Dimensional Array and Double Dimensional Array
- Different Operations on Array
- Difference between Sorting and Searching

Number of Periods	
Theory	Practical
3	2

Teaching Plan

Before starting the chapter, make the students to revise about JAVA topics taught in earlier chapters for better understanding of the current topic.

Explain the meaning of arrays in detail with the help of examples to the students. Also, tell the students about Array as a composite type.

Define the types of arrays with the help of suitable examples to the students:

- Single Dimensional Array
- Multi-Dimensional Array

Explain the meaning of length of an array with the help of an example.

Explain how to take input from user, declare a single or multi-dimensional array and assign values to the array in detail with the help of examples and programs for better understanding.

Tell the difference between single dimensional and double dimensional array with the help of following table.

Single Dimensional Array	Double Dimensional Array
Contains single column and multiple rows.	Contains multiple rows and multiple columns.
The variables are having same name with single subscript value.	The variables are having same name with two subscript values one for row and one for column.
To declare the array: int ar[]=new int[10];	To declare the array: int ar[][]=new int[2][4];

Explain the different operations on array which are:

- Searching
- Sorting
- Insertion
- Deletion

- Merging

Explain the difference between the followings in detail and with the help of following tables:

Linear Search	Binary Search
It works on both sorted and unsorted arrays.	It works only on sorted arrays.
It starts searching from 0th position.	It starts searching from the middle element and proceeds either to the left or to the right depending on the value of the searched element.

Bubble Sort	Selection Sort
Compares with the next element and swaps if the condition matches	Selects the minimum element from the unsorted part of the array and places it in the next position in the sorted part of the array.
Execution is slow.	Execution is fast.
Less efficient	More efficient

Sorting	Searching
1. Arranging the array elements in ascending or descending order.	1. Finding an element in the array.
2. Two types of sorting are: Bubble sort and Selection sort.	2. Two types of searching are: Linear search and Binary search.

Extension

Ask the students some oral questions based on this chapter.

Q. What is an array?

Q. What are the types of array?

Q. What is single dimensional array?

Q. What is multi-dimensional array?

Q. What is double dimensional array?

Q. Write the difference between:

a. Sorting and Searching

b. Bubble Sort and Selection Sort

c. Linear Search and Binary Search

Evaluation

After explaining the chapter, let the students do the exercises given on Page 363 to 374 in the main course book as **Mind Drill and Previous Years' Questions**.

16. String Handling

Teaching Objectives

Students will learn about

- ☞ The String Class
- ☞ How String is Stored in Memory?
- ☞ Methods of the String Class
- ☞ String Array

Number of Periods	
Theory	Practical
3	2

Teaching Plan

Before starting the chapter, make the students to revise about JAVA topics taught in earlier chapters for better understanding of the current topic.

Introduce the students with the concept of string and then explain them about the string class and its role.

Tell them about the implicit and explicit string declaration in detail with the help of suitable examples and programs.

Demonstrate the students about how string is stored in memory.

Explain the methods of the string class with the help of suitable examples for better understanding.

Explain what is a string array to the students in detail along with the graphical representation of a string array.

str	0	1	2	3	4	5	6	7	8
0	K	o	l	k	a	t	a		
1	D	e	l	h	i				
2	M	u	m	b	a	i			
3	C	h	e	n	n	a	I		
4	B	e	n	g	a	l	u	r	u

Extension

Ask the students some oral questions based on this chapter.

Q. What is a string?

Q. What is a string class?

Q. What is implicit string declaration?

Q. What is explicit string declaration?

Q. How string is stored in memory?

Q. Explain the methods of the string class.

Q. What is a string array?

Evaluation

After explaining the chapter, let the students do the exercises given on Page 395 to 409 in the main course book as **Mind Drill and Previous Years' Questions**.