TOUCHPAD

Modular Python Ver. 1.0

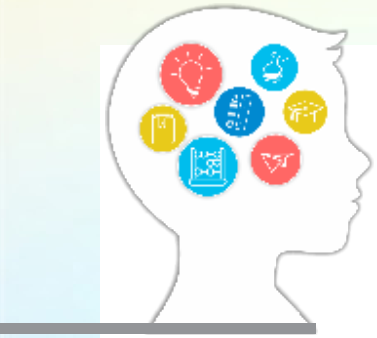
Teacher's Manual

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 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 in 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 need 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.

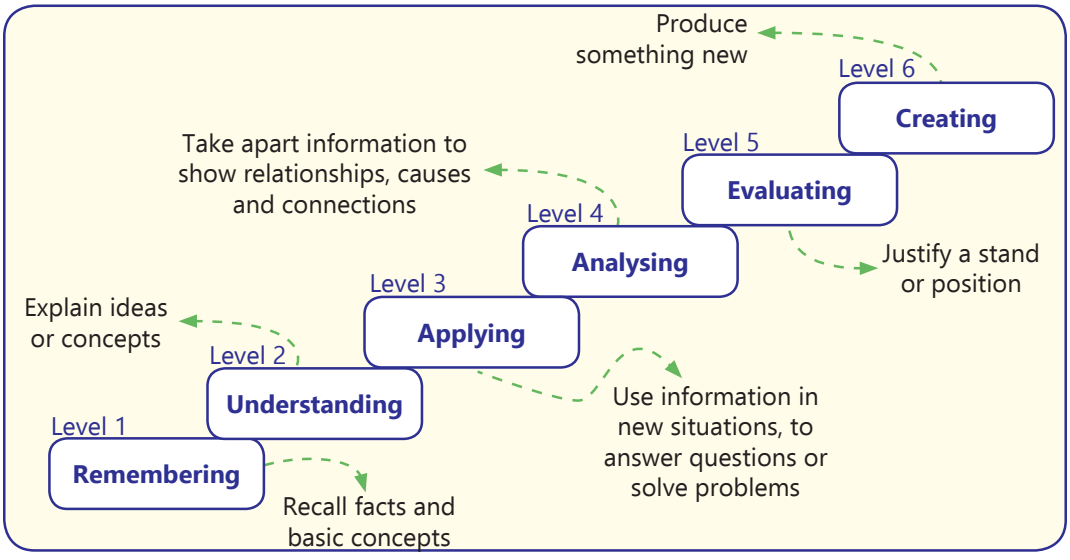
"Knowing yourself is the beginning of all wisdom."

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 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."

LESSON PLAN

Touchpad MODULAR PYTHON Ver 1.0

1. Introduction to Python

Teaching Objectives

Students will learn about

- ☞ Features of Python
- ☞ Programming in Python
- ☞ Variables in Python
- ☞ Installing Python
- ☞ Input and Output in Python

Number of Periods	
Theory	Practical
1	3

Teaching Plan

While teaching this chapter, tell the students that Python is a popular high-level programming language.

Also, give an introduction to Python and its uses to the students.

Explain the features of Python to the students.

Demonstrate the steps to download and install the Python software.

Teach the students about Programming in Python by explaining the two basic programming modes—Interactive Mode and Script Mode.

Demonstrate the steps to start Python and work in Interactive Mode.

Show the steps to create a new file, writing, saving, and running a Python Program to the students.

Ensure that the scope of the Teacher's Corner given at the end of the chapter has been covered.

Ask the students to read the **Clickipedia and Tech Funda** given on pages 10 and 11.

Extension

Ask the students some oral questions based on this chapter.

Q. Define Python

Q. Who developed Python?

Q. What are the features of Python?

Q. Define Interactive Mode

Q. What are the two components of the Python IDLE window?

Q. What is the full form of IDLE?

Q. What do you mean by Script Mode?

Q. Define the print( ) function.

Encourage the students to walk through the chapter and ask them to explain any one topic from the chapter.

Evaluation

After explaining the chapter, let the students do the course book exercises given on Pages 13 and 14 of the main course book as Exercise.

Take the students to the computer lab and let them practice the activity given in In the Lab section on Page 14 in the main course book. This will enhance the abilities of the students and serve as a Subject Enrichment **Activity**.

Suggested Activity

Ask the students to input ten names from the user and print them in the same line with a single space.

2. Data Types and Operators in Python

Teaching Objectives

Students will learn about

- ☞ Data Types
- ☞ Operators
- ☞ Some More Programs
- ☞ Comments in Python
- ☞ Precedence of Operators

Number of Periods	
Theory	Practical
2	3

Teaching Plan

While teaching this chapter, tell the students that data types are used to define the type of value a data can contain.

Tell the students about data types and their four categories which are:

- Numbers
- Sets
- Sequence
- Boolean

Make them understand that **comments** can be used to explain parts of the code.

Explain to the students that Python supports two types of comments: Single line comment and Multiline comment.

Introduce to the students about operators in Python in detail.

Teach them about the types of operators in Python with examples:

- Arithmetic Operators
- Assignment Operators

- Logical Operators
- Relational Operator

Explain the Precedence of operators to the students by using the table given on page 23.

Ensure that the scope of the Teacher's Corner given at the end of the chapter has been covered.

Ask the students to read the **Clickipedia** given on page 23 and run programs given in the book.

Extension

Ask the students some oral questions based on this chapter.

Q. Define the term data types
Q. What do you mean by Precedence of operators?
Q. Define Assignment Operators.
Q. Define comments in Python.
Q. What are the operators?
Q. How many types of comments does Python support?
Q. Which operators are used to evaluate and decide?

Encourage the students to walk through the chapter and ask them to explain any one topic from the chapter.

Evaluation

After explaining the chapter, let the students do the course book exercises given on Pages 25 to 27 of the main course book as an **Exercise**.

Take the students to the computer lab and let them practice the activity given in **In the Lab** section on Page 27 of the main course book. This will enhance the abilities of the students and serve as a Subject Enrichment Activity.

Suggested Activity

Ask the students to make a Word file containing five Python programs using different operators and code them on Python Idle Window to find the output of it. Also, paste the screenshots of the output in Word.

3. Conditional Statements in Python

Teaching Objectives

Students will learn about

- ☞ if Statement
- ☞ Nested if Statement
- ☞ Some More Programs
- ☞ if...else Statement
- ☞ if...elif...else Ladder

Number of Periods	
Theory	Practical
2	3

Teaching Plan

While teaching this chapter, tell the students that Decision making in Python is done using conditional statements which decide the flow of program execution.

Explain all the conditional statements available in Python to the students. Those are:

- if Statement
- if...else Statement
- Nested if Statement
- if...elif...else Ladder

Demonstrate the syntax and flowchart of each conditional statement to the students.

Demonstrate some more programs using conditional to the students.

Ensure that the scope of the Teacher's Corner given at the end of the chapter has been covered.

Ask the students to read the **Clickipedia and Tech Funda** given on pages 29 and 32.

Extension

Ask the students some oral questions based on this chapter.

Q. Which is the simplest conditional statement?

Q. Which statement checks for a condition?

Q. What do you mean by nested if statement?

Q. What are the types of conditional statements available in Python?

Encourage the students to walk through the chapter and ask them to explain any one topic from the chapter. Also, ask them to solve Periodic Assessment-1 given on page no 39.

Evaluation

After explaining the chapter, let the students do the course book exercises given on Pages 36 to 38 of the main course book as an **Exercise**.

Take the students to the computer lab and let them practice the activity given in **In the Lab** section on Page 38 of the main course book. This will enhance the abilities of the students and serve as a Subject Enrichment Activity.

Suggested Activity

Ask the students to collect more information on the topic "Conditional statements" and create a PowerPoint presentation of 10 slides on the same.

4. Looping Statements in Python

Teaching Objectives

Students will learn about

- ☞ The for Statement
- ☞ The jump Statements
- ☞ The while Statement
- ☞ Some More Programs

Number of Periods	
Theory	Practical
2	3

Teaching Plan

While teaching this chapter, tell the students that In Python, the statements that are used to repeat a set of instructions are called iterative or looping statements. Looping statements are very useful and necessary for developing applications.

Explain the syntax and working of the for statement in Python.

Provide examples and guide students in writing and executing for loops with different variations.

Introduce the while statement and its purpose in Python.

Explain the syntax and working of the while statement, including the condition that controls the loop.

Introduce the concept of jump statements, specifically the break and continue statements.

Engage students in coding exercises to practice using for, while and jump statements effectively.

Present additional program scenarios where looping statements can be applied.

Also, provide practical exercises involving nested loops, loop control with user input, or pattern printing using loops.

Ask the students to read the **Clickipedia** given on pages 43 and 47.

Extension

Ask the students some oral questions based on this chapter.

Q. What is the purpose of the for statement in Python?

Q. How does the while statement differ from the for statement?

Q. What is the use of the break and continue statements in loops?

Q. What are the two types of looping statements in Python?

Q. What is the purpose of the for statement in Python?

Q. What is the range() function used for in Python?

Q. How can you define a sequence in Python?

Q. How does the while statement differ from the for statement in terms of execution?

Q. How can you terminate an infinite loop in Python?

Q. When is the else block executed in a while loop?

Q. What is the purpose of the break keyword in Python?

Q. Explain the functionality of the continue statement in a loop.

Evaluation

After explaining the chapter, let the students do the course book exercises given on pages 48 and 49 of the main course book as an **Exercise**.

Take the students to the computer lab and let them practice the activity given in **In the Lab** section on page 49 of the main course book. This will enhance the abilities of the students and serve as a Subject Enrichment Activity.

Suggested Activity

Ask the students to design and implement a program using looping statements to solve a specific problem or perform a task of their choice. They can present their programs to the class and explain their approach and outcomes. This activity promotes creativity, independent thinking, and practical application of looping concepts in Python.

5. Functions in Python

Teaching Objectives

Students will learn about

- ☞ Features of Function
- ☞ Advantages of Functions
- ☞ Calling a Function
- ☞ Components of Python Function
- ☞ Types of Functions in Python Creating a Function
- ☞ Some More Programs

Number of Periods	
Theory	Practical
2	4

Teaching Plan

While teaching this chapter, tell the students that A function is a block of organized and reusable code used to perform a single or related action.

Introduce the topic of functions in Python and their significance in code organization and reusability to the students.

Explain the key features of functions to the students.

Discuss the components of a Python function to the students. Those are:

- Name of the function
- Arguments
- Statements
- Return Value

Let them know the advantages of using functions in Python.

Introduce different types of functions in Python, such as built-in functions and user-defined functions

Discuss the steps involved in creating a function in Python.

Explain the concept of function invocation or calling a function to the students.

Present additional program scenarios where functions can be applied.

Provide practical exercises involving function calls within loops, functions with multiple return values, or functions with optional arguments.

Ask the students to read the **Tech Funda** given on pages 51

Extension

Ask the students some oral questions based on this chapter.

Q. What are the advantages of a function?

Q. What are built-in functions?

Q. Define arguments in Python.

Q. Which keyword is used to define the functions?

Q. What do you mean by function?

Q. What are the components of a Python function?

Encourage the students to walk through the chapter and ask them to explain any one topic from the chapter. Also, ask them to solve Periodic Assessment-2 and Test Sheet-1 given on pages no 57 to 59.

Evaluation

After explaining the chapter, let the students do the course book exercises given on pages of the main course book as an **Exercise**.

Take the students to the computer lab and let them practice the activity given in **In the Lab** section on page of the main course book. This will enhance the abilities of the students and serve as a Subject Enrichment Activity.

Suggested Activity

Ask the students to collect more information on the topic "Functions in Python" and create a PowerPoint presentation of 10 slides on the same.

6. String Handling in Python

Teaching Objectives

Students will learn about

- ☞ String
- ☞ String Operators
- ☞ Some More Programs
- ☞ Traversing a String
- ☞ Python Built-in Functions

Number of Periods	
Theory	Practical
2	3

Teaching Plan

While teaching this chapter, tell the students that sequence is an ordered collection of items, which are indexed by positive integers.

Introduce the topic of string operations and their significance in Python programming.

Explain how to traverse or iterate over a string using loops.

Discuss various string operators, including concatenation, comparison, and membership operators.

Introduce important Python built-in functions for string manipulation, such as len(), lower(), upper(), capitalize() and count().

Present additional program scenarios where strings can be applied.

Provide practical exercises involving string searching, pattern matching, or string formatting.

Extension

Ask the students some oral questions based on this chapter.

Q. What are some important string operators in Python?

Q. What is a string?

Q. Define traversing in a string?

Q. Which functions are used to modify and manipulate strings?

Q. Which operator is used to repeat the string for a given number of times?

Q. Which function returns a string with the first character in capital?

Evaluation

After explaining the chapter, let the students do the course book exercises given on pages 67 and 68 of the main course book as an **Exercise**.

Take the students to the computer lab and let them practice the activity given in **In the Lab** section on page 68 of the main course book. This will enhance the abilities of the students and serve as a Subject Enrichment Activity.

Suggested Activity

Ask the students to design and implement a program that takes a user-input string and performs the following operations:

- Count the number of vowels and consonants in the string.
- Reverse the string and check if it is a palindrome.
- Replace all occurrences of a specified character with another character.

7. List in Python

Teaching Objectives

Students will learn about

- ☞ Creating a List
- ☞ Operations on a List
- ☞ Python Functions
- ☞ Changing the List Element
- ☞ Some More Programs
- ☞ Traversing a List
- ☞ List Methods
- ☞ Slicing the List
- ☞ Adding an Element to a List

Number of Periods	
Theory	Practical
2	3

Teaching Plan

While teaching this chapter, tell the students that a list is a sequence of multiple values in a sequence. In a list, each element or value is called an item.

Explain lists to the students and how they are used to store multiple elements in Python.

Show them examples of creating lists with different types of elements.

Teach various methods for iterating through a list, such as using for loops and while loops.

Demonstrate how to access and process each element in the list during traversal.

Teach various ways to traverse a list such as:

- Indexing
- Negative Indexing
- Slicing

Introduce common operations performed on lists to the students. Those are:

- Joining lists
- Repeating lists
- Slicing lists
- Comparing list

Introduce them with built-in list methods which are append(), extend(), insert(), remove(), index( ), count( ), pop([i]), copy( ), clear( ), sort( ) and reverse( ).

Provide practical examples to illustrate the usage of each method and their respective outcomes.

Explain the concept of functions in Python to the students.

Explain slicing as a way to extract a portion of a list.

Provide examples to demonstrate common slicing techniques.

Show them how to change the elements in a list.

Demonstrate the use of the append() and extend() methods to add elements to a list.

Provide students with practical programming challenges.

Ask the students to read the **Clickipedia** given on pages 70, 71 and 73.

Extension

Ask the students some oral questions based on this chapter.

Q. What is a list in Python?

Q. What does "traversing" mean in the context of a list?

Q. How can you access elements in a nested list?

Q. What does negative indexing mean in relation to a list?

Q. Is a list mutable or immutable?

Q. How can you change elements or add new elements to a list?

Q. Name some operations that can be performed on lists.

Q. What are some of the built-in list methods in Python?

Q. List three built-in list functions in Python.

Encourage the students to walk through the chapter and ask them to explain any one topic from the chapter. Also, ask them to solve Periodic Assessment-3 given on page no 79.

Evaluation

After explaining the chapter, let the students do the course book exercises given on pages 77 and 78 of the main coursebook as an **Exercise**.

Take the students to the computer lab and let them practice the activity given in **In the Lab** section on page 78 of the main course book. This will enhance the abilities of the students and serve as a Subject Enrichment Activity.

Suggested Activity

Ask the students to plan a virtual trip around the world using a list of destinations, expenses, and activities for a budget-friendly adventure.

8. Tuple in Python

Teaching Objectives

Students will learn about

- Creating a Tuple
- Unpacking Tuple
- Python Tuple Methods
- Operations on a Tuple
- Deleting a Tuple
- Some More Programs
- Accessing a Value in a Tuple
- Traversing Elements in a Tuple
- Python Functions
- Changing a Tuple
- Difference between List and Tuple

Number of Periods	
Theory	Practical
2	3

Teaching Plan

While teaching this chapter, tell the students that

Ask the students to read the **Clickipedia** given on pages 81 and 85.

Extension

Ask the students some oral questions based on this chapter.

Q. What is a tuple in Python?

Q. How would you describe the order of elements in a tuple?

Q. Can you modify the elements of a tuple after it's created? Why or why not?

Q. What does "traversing" mean in the context of a tuple?

Q. How do you access elements in a tuple using indexing?

Q. What is the index of the last element in a tuple?

Q. How do you use slicing to access a range of elements in a tuple?

Q. Can individual elements be deleted from a tuple?

Q. How can you delete the entire tuple?

Q. How would you print the entire contents of a tuple?

Encourage the students to walk through the chapter and ask them to explain any one topic from the chapter.

Evaluation

After explaining the chapter, let the students do the course book exercises given on pages 88 to 90 of the main course book as an **Exercise**.

Take the students to the computer lab and let them practice the activity given in **In the Lab** section on page 90 of the main course book. This will enhance the abilities of the students and serve as a Subject Enrichment Activity.

Suggested Activity

Ask the students to imagine that they are organizing a class picnic. Create a tuple 'picnic_items' with the essential items needed, such as sandwiches, drinks, and snacks. Display the third item from the tuple in your presentation.

9. Dictionary in Python

Teaching Objectives

Students will learn about

- ☞ Creating a Dictionary
- ☞ Python Methods
- ☞ Updating a Dictionary
- ☞ Some More Programs
- ☞ Accessing a Dictionary
- ☞ Python Functions
- ☞ Removing or Deleting Elements of a Dictionary

Number of Periods	
Theory	Practical
2	3

Teaching Plan

While teaching this chapter, tell the students that a dictionary in Python is another data type that contains a collection of values in the form of key-value pairs.

Define what a dictionary is in Python and explain that it stores data in key-value pairs.

Provide real-world examples to illustrate the concept of dictionaries.

Demonstrate how to create a dictionary using the {} notation and key-value pairs.

Show examples of dictionaries with various data types as values.

Explain how to access elements in a dictionary using their keys.

Explain various Python methods to the students with examples.

Let them know the built-in functions. Some of them are:

- sorted( )
- len( )
- cmp( )
- any( )
- all( )

Discuss with them how to update a dictionary using in Python.

Teach them how to remove or delete elements of a dictionary.

Provide programming exercises where students can practice creating, accessing, and updating dictionaries.

Extension

Ask the students some oral questions based on this chapter.

Q. How can a Dictionary be created in Python?

Q. Which function(s) can be used to delete specific values from a Dictionary?

Q. How many elements can Python Dictionaries be defined into?

Q. How can you retrieve or access a specific data element in a dictionary?

Q. What is the method for accessing elements in a dictionary using the associated key: value pairs?

Evaluation

After explaining the chapter, let the students do the course book exercises given on pages 95 and 96 of the main coursebook as an **Exercise**.

Take the students to the computer lab and let them practice the activity given in **In the Lab** section on page 96 of the main course book. This will enhance the abilities of the students and serve as a Subject Enrichment Activity.

Suggested Activity

Ask the students to create a PowerPoint presentation on the topic "Functions in Python".

10. App Development

Teaching Objectives

Students will learn about

- ☞ What is an App?
- ☞ Types of Mobile Apps
- ☞ Downloading and Installing the App
- ☞ Android and iOS
- ☞ Categories of Apps
- ☞ Developing an App

Number of Periods	
Theory	Practical
2	1

Teaching Plan

While teaching this chapter, tell the students that an app is a software program primarily developed for hand-held smart devices such as mobile and tablets.

Ask students if they have used any apps and what they like about them.

Define what an app is and explain that it is a software application designed for mobile devices.

Explain the difference between Android and IOS.

Discuss the various purposes of apps, such as productivity, entertainment, education, etc.

Introduce the two main types of mobile apps: Native Apps, Web Apps and Hybrid Apps.

Present different categories of mobile apps, such as Gaming Apps, Productivity Apps, Entertainment Apps, Utility Apps, Educational Apps, Social Networking Apps, Communication Apps and E-Commerce Apps.

Show examples of popular apps from each category.

Demonstrate how to download and install apps from the app stores (Google Play Store for Android and Apple App Store for iOS).

Discuss the importance of downloading apps from trusted sources.

Introduce the concept of app development, where developers create software applications.

Teach the steps to create an app using App Inventor. The steps are given below:

Step 1: Type URL: https://AppInventor.mit.edu and press **Enter** key.

Step 2: Click on **Create Apps!** Button.

Step 3: Type your user Gmail username and password.

Step 4: Click on Next button.

Step 5: Click on I accept the terms of service! Button.

Step 6: Click on **Continue** to create an app.

Step 7: Click on **Start new project** button to start your first project.

Step 8: Type the name for your project in the **Project name:** box.

Step 9: Click on **OK** button.

Ask the students to read the **Clickipedia** given on page 102

Extension

Ask the students some oral questions based on this chapter.

Q. What is the primary purpose of an app?

Q. Name three popular app stores for different operating systems.

Q. What are the three main types of applications based on the platform they run on?

Q. Which operating system is developed by Google for mobile devices?

Q. What is the operating system developed by Apple Inc. for iPhones and iPads called?

Q. Name the three types of mobile apps based on their development approach.

Q. Which platform can be used to create apps for smartphones and is available for free?

Q. What do you understand by the term App?

Q. What are Web apps?

Q. What is a native mobile app?

Encourage the students to walk through the chapter and ask them to explain any one topic from the chapter. Also, ask them to solve Periodic Assessment-4 given on pages 110 to 112.

Evaluation

After explaining the chapter, let the students do the course book exercises given on pages 108 and 109 of the main coursebook as an **Exercise**.

Take the students to the computer lab and let them practice the activity given in **In the Lab** section on page 109 of the main course book. This will enhance the abilities of the students and serve as a Subject Enrichment Activity.

Suggested Activity

Ask the students to research and present an app from a specific category, including its purpose, features, and target audience.