

Unit 03

Programming Fundamentals

EXERCISE

Multiple Choice Questions (MCQs)

MCQ	1	2	3	4	5	6	7	8	9
Answer	a	c	c	d	c	b	c	b	d
MCQ	10	11	12	13	14	15	16	17	18
Answer	d	a	c	a	c	b	c	a	c

Short Response Questions (SRQs)

Q1. What are the applications of computer programming in daily life?

Computer programming has become an integral part of our daily lives, influencing a vast array of activities and industries. Here are some of the primary applications:

- Social Media Platforms like Facebook, Twitter, Instagram, and LinkedIn are powered by complex algorithms and code that enable us to connect with friends, family, and professionals globally.
- Gmail, Outlook, and other email services rely on programming to manage the sending, receiving, and organizing of emails efficiently.
- Banking apps and websites use programming to facilitate transactions, track account balances, and ensure secure access to financial information.

Q2. Write code to take input a number from user and print its mathematical table on screen from 1 to 10.

Program:

```
x = eval(input('Enter a number to get its table:'))
for i in range(10):
    print(x, 'x', i+1, '=', x*(i+1))
```

Q3. Take an odd number as input from the user, check if it is odd, otherwise ask the user to reenter an odd number.

while True:

```
x=eval(input('Enter an Odd Number:'))
if (x%2 != 0):
    print("You Entered an Odd Number")
    break
else:
    print("Number is Even, Re Enter the Number")
```

Q4. Write down the main examples of Python based application.

Following are some examples of python-based applications:

- i. Web Development
 - a. Django: Instagram, Pinterest.
 - b. Flask: LinkedIn, Netflix.
- ii. Education
 - a. Jupyter Notebooks: Teaching and research.
 - b. Code Combat: Learning through games.
- iii. Data Science
 - a. Pandas, NumPy: Data manipulation.
 - b. Matplotlib, Seaborn: Visualization.
- iv. Machine Learning
 - a. TensorFlow, PyTorch: Deep learning.
 - b. Scikit-learn, Keras: ML libraries.

Q.5 Differentiate between global and local variables with the help of suitable example.

	Local Variable	Global Variable
Scope	Accessible only within the function where it is defined.	Accessible throughout the entire program, both inside and outside of functions.
Declaration	Declared within a function.	Declared outside any function.
Access	Can be modified directly within the function but cannot be accessed or modified outside of it.	Can be modified inside a function using the 'global' keyword.
Example	<pre>def my_function(): y = 5 # Local variable print("Value of y:", y)</pre>	<pre>x = 10 # Global variable def my_function(): print("Inside function, x:", x) my_function() print("Outside function, x:", x)</pre>

Extended Response Questions (ERQs)

Q1. Explain the applications of Python in different business and technical domains.

Python is a versatile programming language widely used across various business and technical domains due to its simplicity, readability, and extensive library support. Here are some key applications:

- i. Web Development
 - a. Frameworks: Django, Flask, Pyramid.
 - b. Use: Building robust websites, web applications.
 - c. Examples: Instagram, Pinterest, and Spotify use Django.
- ii. Data Science and Analytics
 - a. Libraries: Pandas, NumPy, SciPy.
 - b. Use: Data manipulation, statistical analysis, data cleaning.
 - c. Examples: Financial analysis, market research, data visualization.
- iii. Machine Learning and Artificial Intelligence
 - a. Libraries: TensorFlow, PyTorch, Scikit-learn, Keras.

- b. Use: Predictive analytics, natural language processing, image recognition.
- c. Examples: Recommendation systems, fraud detection, autonomous vehicles.
- iv. Game Development
 - a. Libraries: Pygame, Panda3D.
 - b. Use: Creating simple to moderately complex games.
 - c. Examples: Prototyping game ideas, educational games, small indie games.
- v. Desktop Applications
 - a. Libraries: Tkinter, PyQt, Kivy.
 - b. Use: Building cross-platform desktop applications with graphical user interfaces.
 - c. Examples: Text editors, file management tools, custom utilities.
- vi. Blockchain Development
 - a. Libraries: Web3.py, Brownie.
 - b. Use: Interacting with blockchain networks, developing and testing smart contracts.

Q2. What are the basic functions that 'List' provides? Elaborate each of them with an example.

Len() Function

It provides with the total number of elements in a list

Min() Function

It is used to extract the smallest number from the elements in a list

Max() Function

It is used to extract the largest number from the elements in a list.

Sum() Function

It sums up all the elements of the list.

Insert() Function

It inserts an element in the list.

Index() Function

It returns the position of a number in a list. In case of multiple occurrences of the number, the placement of first match is returned.

Randint() Function

It is used to generate a random number in the given range.

Q3. Write a program for Dice Rolling Race Game for 2 Players.

Python Code

```
import random

target = 100
p1,p2 = 0,0

while p1 < target and p2 < target:
    p1 += random.randint(1, 6)
    print("P1: {p1}")
```

```
if p1 >= target:
    print("P1 wins!")
    break

p2 += random.randint(1, 6)
print(f"P2: {p2}")

if p2 >= target:
    print("P2 wins!")
    break
```

Q.4 How to locate and select Debugger in IDLE? Write steps by taking an example into consideration.

Step1: Open “Python IDLE” from start menu.

Step2: Write your computer program using python language.

Step3: From the IDLE menu Bar, Click on Debug option, Select “Debugger” from the drop-down menu.

Step4: A window titled as “Debug Control” will pop-up on screen.

Step5: Notice a “Debug On” message on the shell prompt.

Step6: Watch variables and their corresponding values.

Q5. Write code to print the multiplication of first 10 odd numbers and first 10 even numbers and find the difference of the two using functions.

Program

```
def prod_of_odds(n):
    product=1
    for i in range(1,2*n,2):
        product*=i
    return product

def prod_of_evens(n):
    product=1
    for i in range (2,2*n+1,2):
        product*=i
    return product

#calculate products
odd_product = prod_of_odds(10)
even_product = prod_of_evens(10)

#calculate the difference
difference = even_product - odd_product
```

#print the results

```
Print('Product of first 10 odd numbers:', odd_product)
Print('Product of first 10 even numbers:', even_product)
Print('Difference:', difference)
```

Mini projects

Mini Project 1: Number Guessing Game

Generate a Random Number and ask the user to guess the number.

- For each input respond the user, whether the guess is right or wrong.
- If the guess was right, congratulate the user and inform about how many guesses it took till the right guess.
- However for each wrong answer, inform the user to try again. Accordingly, also provide a hint whether the guess was on the higher or lower side.
- Give the user appropriate number of tries e.g. if the guess is between 1 to 20, you may provide with 3 retries.

Program

```
import random
def number_game():
    number=random.randint(1,100)
    attempts=0
    max_attempts=3
    print("Guess the number between 1-100.")

    while attempts<max_attempts:
        guess= input(f "Attempts {attempts+1}/{max_attempts}: Your guess?")

        if not guess.isdigit():
            print("Invalid input. Retry")
            continue

        guess=int(guess)
        attempts+=1

        if guess<1 or guess>100:
            print("Please guess a number between 1-100")
        elif guess<number:
            print("Too low! Try again.")
        elif guess>number:
            print("Too high! Try again")
        else:
            print(f "Congratulations! You guessed the number in {attempts}
attempts.")

    if attempts==max_attempts and guess!=number:
        print(f "sorry! You have used all attempts. The number was {number}.")
```

Mini Project 2: Motorcycle Rent Calculator

Motorcycles can be rented either based on distance travelled or the time the motorcycle was being used by the client. Both are separately calculated and added together, to generate total bill.

- For example, hourly rate is Rs. 100 while the distance rate is Rs. 5 / km.
- A client travelled only 20 km but the possession of the motorcycle was for 3 hours. This way the bill of travelled distance will be Rs. 100/- but the possession-charges will be Rs. 300. this way total bill will be of Rs. 400.
- However for existing members, the lesser of the two charges is treated as a “*discount*”.
- This way if the client in above example was already a member, only Rs. 300/- needs to be paid by him.

Program

```
def rent_calculator(kms, hours, is_member):  
    #Rates  
    hourly_rate=100  
    distance_rate=5  
  
    #calculate the charges  
    distance_charge = kms * distance_rate  
    time_charge = hours * hourly_rate  
    total_charge = distance_charge + time_charge  
  
    #Apply discount for members  
    if is_member:  
        total_charge= min(distance_charge, time_charge)  
    return total_charge  
  
#Example  
kms=20  
hours=3  
is_member= True  
  
#calculate rent  
total_bill= rent_calculator(kms, hours, is_member)  
print('Total bill: Rs.',total_bill)
```