Here are the answers to the true/false questions:

1. False

2. True

3. False

4. True

5. False

6. True

7. True

8. False

9. True

10. False

answers to the multiple-choice questions:

1. c) Fee setting

2. a) F = 9/5(C) + 32

3. d) specification

4. c) 2spam

5. b) statements

6. b) expressions

7. b) Program

8. a) a general for loop

9. b) variable-as-box

10. d) input

1. The Six Steps in the Software Development Process:

- Specification: Define the problem and what the software should accomplish.

- Design: Plan the architecture of the software, including data structures and interfaces.

- Implementation: Write the actual code based on the design specifications.

- Testing/Debugging: Run tests to find and fix bugs or errors in the code.

- Maintenance: Update the software to fix issues, improve performance, or add new features after deployment.

- Documentation: Create user manuals and technical documentation to help users and developers understand the software.

2. Chaos.py Program:

Import random

def main():

for i in range(100):

x = random.random()

y = random.random()

print(x, y)

main()

The Identifiers include the following in the program:

- main

- i

- x

- y

- random

- print

Underline Each Expression

Here’s a modified version of the chaos.py program with expressions underlined and comments indicating the type of statement:

import random # import statement

def main(): # function definition

for i in range(100): # loop

x = random.random() # assignment

y = random.random() # assignment

print(x, y) # output

main() # function call

Relationships Among Concepts

- Definite Loop: A loop that iterates a specific number of times. It often uses a range of values.

- For Loop: A type of definite loop in Python that iterates over a sequence (like a list or range).

- Counted Loop: Another term often used for a for loop when the number of iterations is predetermined (i.e., counting the number of times to run the loop).

Output from Fragments

what each fragment would output:

(a)

for i in range(5):

print(i \* i)

Output:

0

1

4

9

16

(b)

for d in [3, 1, 4, 1, 5]:

print(d, end=" ")

Output:

3 1 4 1 5

(c)

for i in range(4):

print("Hello")

Output:

Hello

Hello

Hello

Hello

(d)

for i in range(5):

print(i, 2\*\*i)

Output:

0 1

1 2

2 4

3 8

4 16

Writing Out an Algorithm in Pseudocode

Writing an algorithm in pseudocode before coding helps clarify the logic and flow of the program. It allows you to:

- Identify potential errors early in the design phase.

- Focus on the logic without getting bogged down in syntax.

- Facilitate communication with others who may help or review your work.

The `sep` Parameter

The `sep` parameter in the `print` function specifies a string that separates multiple arguments. For example:

print("Hello", "World", sep=", ")

Output:

Hello, World

Prediction of the Code Execution

print("start")

for i in range(0):

print("Hello")

print("end")

Output:

start

end

The loop will not execute because `range(0)` produces an empty range. Thus, only "start" and "end" will be printed.

1. Modify the `convert.py` Program:

print("This program converts Celsius to Fahrenheit.")

2. Modify the `avg2.py` Program:

score1 = float(input("Enter first exam score: "))

score2 = float(input("Enter second exam score: "))

score3 = float(input("Enter third exam score: "))

average = (score1 + score2 + score3) / 3

print("The average score is:", average)

3. Modify the `convert.py` Program with a Loop:

for \_ in range(5):

celsius = float(input("Enter temperature in Celsius: "))

fahrenheit = (9/5) \* celsius + 32

print("Fahrenheit:", fahrenheit)

4. Celsius to Fahrenheit Table

Modify `convert.py` to compute and print a table of Celsius temperatures and their Fahrenheit equivalents every 10 degrees from 0°C to 100°C.

print("Celsius to Fahrenheit Conversion Table")

print("Celsius\tFahrenheit")

for celsius in range(0, 101, 10):

fahrenheit = (celsius \* 9/5) + 32

print(f"{celsius}\t{fahrenheit}")

5. User Input for Investment Years

# Get user input

principal = float(input("Enter the amount to invest: "))

rate = float(input("Enter the annual interest rate (as a percentage): ")) / 100

years = int(input("Enter the number of years: "))

# Calculate future value

future\_value = principal \* (1 + rate) \*\* years

# Print result

print(f"After {years} years, your investment will be worth: ${future\_value:.2f}")

6. Total Accumulation of Investment

# Get user input

annual\_investment = float(input("Enter the amount to invest each year: "))

rate = float(input("Enter the annual interest rate (as a percentage): ")) / 100

years = int(input("Enter the number of years: "))

# Calculate total accumulation

total\_amount = 0

for year in range(years):

total\_amount = (total\_amount + annual\_investment) \* (1 + rate)

# Print result

print(f"After {years} years, your total accumulation will be: ${total\_amount:.2f}")

7. Nominal Rate and Compounding Periods

# Get user input

principal = float(input("Enter the initial investment amount: "))

rate = float(input("Enter the annual interest rate (as a percentage): ")) / 100

periods = int(input("Enter the number of times interest is compounded per year: "))

years = 10

# Calculate future value with compounding

future\_value = principal

for \_ in range(years \* periods):

future\_value \*= (1 + rate / periods)

# Print result

print(f"After {years} years, your investment will be worth: ${future\_value:.2f}")

8. Fahrenheit to Celsius Conversion

A program to convert temperatures from Fahrenheit to Celsius

# Get user input

fahrenheit = float(input("Enter temperature in Fahrenheit: "))

celsius = (fahrenheit - 32) \* 5/9

print(f"{fahrenheit}°F is {celsius:.2f}°C.")

9. Kilometers to Miles Conversion

A program to convert kilometers to miles.

# Get user input

kilometers = float(input("Enter distance in kilometers: "))

miles = kilometers \* 0.62

print(f"{kilometers} kilometers is approximately {miles:.2f} miles.")

#10. Custom Unit Conversion Program

A program to perform a unit conversion (e.g., pounds to kilograms).

print("This program converts pounds to kilograms.")

# Get user input

pounds = float(input("Enter weight in pounds: "))

kilograms = pounds \* 0.453592

print(f"{pounds} pounds is approximately {kilograms:.2f} kilograms.")

11. Interactive Calculator Program

print("Welcome to the interactive calculator! Type 'exit' to quit.")

while True:

expression = input("Enter a mathematical expression: ")

if expression.lower() == 'exit':

break

try:

result = eval(expression)

print(f"The result is: {result}")

except Exception as e:

print("Invalid expression. Please try again.")