

**FINAL EXAMINATION****BACHELOR OF INFORMATION TECHNOLOGY (HONOURS) IN COMPUTER
APPLICATION DEVELOPMENT****COURSE : PYTHON PROGRAMMING LANGUAGE****COURSE CODE : SWC3643****DURATION : 3 HOURS****INSTRUCTIONS TO CANDIDATES:**

1. This question paper consists of **SIX (6)** questions.
2. Answer ALL questions in Answer Booklet provided.
3. Please check to make sure that this examination pack consists of:
 - i. The Question Paper
 - ii. An Answer Booklet
4. Do not bring any material into the examination hall.
5. Please write your answer using permanent ink.

MYKAD/ PASSPORT NO. : _____

ID. NO. : _____

LECTURER : _____

SECTION : _____

DO NOT OPEN THIS QUESTION PAPER UNTIL YOU ARE TOLD TO DO SO

This question paper consists of 6 printed pages including the front page

QUESTION 1

- a. List **THREE (3)** features that show Python is a **high-level** programming language.
(3 marks)
- b. Python is widely used in many domains such as web development, automation, data analytics and AI. Identify **TWO (2)** reasons why Python is popular across different industries.
(2 marks)
- c. Explain **TWO (2)** Python's main **design philosophy** that influences how Python code is written and read.
(2 marks)
- d. Name **ONE (1)** Python development/distribution platform and describe **one scenario** where it is most suitable to be used.
(3 marks)
- e. Evaluate the following Python expression and show the final result:

`result = (12 // 5) + (7 % 3) * 2 - (3 ** 2)`

(5 marks)
- f. Python uses `bool` type name to represent logical values.
- i. Explain how Boolean values are produced and used in Python
(2 marks)
- ii. Write a Python code segment that asks the user to enter an integer representing the number of hours they sleep per day, and prints `True` only if the value is **valid** (0 to 24); otherwise print `False`.
(3 marks)

(Total : 20 marks)

QUESTION 2

- a. Explain **TWO (2)** differences between a single if statement and a nested if structure in Python.

Then, describe one situation where using a nested if structure is more appropriate than a single if statement.

(6 marks)

- b. Consider the scenario below.

A training centre offers **financial assistance** to participants based on their **assessment score** and **employment status**.

The eligibility rules are as follows:

- If a participant's **assessment score is below 60**, no assistance is given.
- If the score is **60 or above**
- If the participant is **unemployed**, they receive **RM1,200**.
- If the participant is **employed**
- If their **monthly salary is RM3,000 or below**, they receive **RM800**

- i. Write a Python code segment that accepts the participants :

1. Assessment score
2. Employment status / salary

(2 marks)

- ii. Write a Python code segment that determines the **assistance amount** the participant is eligible for based on the rules above.

(9 marks)

- iii. Write a Python code segment that displays the **final assistance amount** in **Ringgit Malaysia (RM)**, formatted to **two decimal places**.

(3 marks)

(Total: 20 marks)

QUESTION 3

- a. Describe the concept of a **non-terminating loop** and explain one common programming mistake that can cause it. Provide a Python code example and describe one technique to ensure the loop ends correctly.

(5 marks)

- b. In Python, `continue` is used as one of the control flow statements.

- i. Explain the purpose of the `continue` statement in Python and how it affects loop execution.

(2 marks)

- ii. Write a Python loop that **skips a specific value** during iteration using the `continue` statement.

(4 marks)

- c. Explain the Python fragment below:

```
num = 1
while num < 5:
    print(num)
    num += 1
```

(2 marks)

- d. Consider the scenario below.

A subscription service charges users based on the **number of weeks** they subscribe.

The system should:

- Continuously request the user to enter the number of weeks,
- Accept the input **only if it is greater than zero**,
- Apply a **discount** if the subscription is **8 weeks or more**, and
- Display the **final subscription cost**.

Write a Python program using a `while` loop to perform the above task.

(7 marks)

(Total : 20 marks)

QUESTION 4

- a. Explain the purpose of **tuples** and **dictionaries** in Python's data structures. (4 marks)
- b. Write a **while loop** that counts from **5 to 25** and prints **only numbers that are divisible by 5**. (5 marks)
- c. Consider the scenario below.

A café uses a simple order processing system to manage customer orders received at the counter. Each order is stored in a list representing the sequence in which the orders are received. Some orders may be marked as cancelled due to customer changes or system updates.

The list of customer orders is given as follows:
latte, espresso, cancelled, muffin, priority, tea, croissant

- i. Write a **for loop** that iterates through the following **list of orders** and prints each order item to the screen.

latte, espresso, cancelled, muffin, priority, tea, croissant

(5 marks)

- ii. The café can only prepare a **maximum of four (4) valid orders** per serving batch.

Write a Python program that:

1. Uses a **for loop** to process the list of orders,
2. Skips any order labelled "cancelled" using the **continue statement**,
3. Stops processing once **FOUR (4)** valid orders have been selected using a **counter and the break statement**, and
4. Prints **only the orders that will be prepared**.

(6 marks)

(Total : 20 marks)

QUESTION 5

- a. Explain the purpose of a **function** and a **module** in Python, and describe how they support code reusability. (4 marks)
- b. Using an appropriate Python module, write a function named `generate_ticket_id()` that returns a random ticket ID between **5000** and **9000**. (5 marks)
- c. A pet supply store sells several types of pet accessories. The store uses a simple program to calculate the total cost of items purchased based on the accessory type and quantity bought. Each transaction is also assigned a random transaction reference number.

The fixed price list is given below:

```
accessory_prices = {  
    'collar': 18.00,  
    'leash': 22.50,  
    'bowl': 12.00  
}
```

- i. Add **exception handling** to your program to manage the following situations:
- Invalid accessory name (not found in price list)
 - File write error
 - Any other unexpected error
- Display an appropriate error message for each case. (5 marks)
- ii. Extend your program by adding **file input/output** features:
- Append each transaction to a file named `transaction_log.txt`
- Output:
- ```
collar, 2, 36.00, 3478
```
- Each transaction must be saved in a **new line**.
  - Use with `open(..., 'a')` to write into the file.
- (6 marks)

(Total : 20 marks)

(TOTAL :100 MARKS)

END OF QUESTION PAPER