

1.write a programme to accept percentage from the user amd display the grade according to the following criteria

- marks>90 | Grade A
- marks>80 and <=90 | Grade B
- marks>=60 and <=80 | Grade C
- Below 60 | Grade D

In [23]:

```
percentage = float(input("Enter the percentage: "))

if percentage > 90:
    grade = "A"
elif percentage > 80 and percentage <= 90:
    grade = "B"
elif percentage >= 60 and percentage <= 80:
    grade = "C"
else:
    grade = "D"

print("Grade: " + grade)
```

Enter the percentage: 65
Grade: C

2.Write a program to accept the cost price of a bike and display the road tax to be paid accroding to the following criteria:

Tax Cost price(in Rs)

- 15% >100000
- 10% >50000 and <=100000
- 5% <=50000

In [24]:

```
cost_price = float(input("Enter the cost price of the bike (in Rs): "))

if cost_price > 100000:
    road_tax = cost_price * 0.15
elif cost_price > 50000:
    road_tax = cost_price * 0.1
else:
    road_tax = cost_price * 0.05

print("The road tax to be paid is Rs", road_tax)
```

Enter the cost price of the bike (in Rs): 50000
The road tax to be paid is Rs 2500.0

3.Accept any city from the user and display monuments of that city.

City Monuments

- delhi Red Fort
- Agra taj Mahal
- Jaipur Jal Mahal

In [31]:

```
city = input("Enter a city name: ")

if city.lower() == "delhi":
    print("Monuments in Delhi:")
    print("- Red Fort")
elif city.lower() == "agra":
    print("Monuments in Agra:")
    print("- Taj Mahal")
elif city.lower() == "jaipur":
    print("Monuments in Jaipur:")
    print("- Jal Mahal")
else:
    print("Sorry, we don't have information about monuments in that city.")
```

Enter a city name: jaipur
Monuments in Jaipur:
- Jal Mahal

4.check how many times a given number can be divided by 3 before it is less than or equal to 10.

In [34]:

```
num = int(input("Enter a number: "))

count = 0
while num > 10:
    num /= 3
    count += 1

print("The number can be divided by 3->:", count, "times before it becomes less than or equal to 10.")
```

Enter a number: 45
The number can be divided by 3->: 2 times before it becomes less than or equal to 10.

5. Why and when to use while loop in python give a detail description with example.

In Python, a while loop is used to execute a block of code repeatedly until a specified condition is met. The syntax of a while loop is as follows:

In []:

```
while condition:
    # code to be executed
```

Here, condition is an expression that is evaluated before each iteration of the loop. If the condition evaluates to True, the code inside the loop is executed. If the condition is False, the loop terminates, and the program continues executing from the next statement after the loop.

A while loop is useful when you want to execute a block of code repeatedly, as long as a certain condition is true. Here are a few scenarios where you might use a while loop in Python:

- 1. When you want to iterate over a sequence of values until a certain condition is met:

In []:

```
# Print all the numbers from 1 to 10
i = 1
while i <= 10:
    print(i)
    i += 1
```

In this example, the loop executes as long as i is less than or equal to 10. Inside the loop, the value of i is printed, and then incremented by 1. This continues until i becomes 11, at which point the condition is no longer true, and the loop terminates.

- 2. When you want to perform an operation on each item in a list until a certain condition is met:

In []:

```
# Multiply all the numbers in a list until the product is greater than 1000
numbers = [2, 3, 5, 7, 11, 13, 17]
product = 1
i = 0
while product <= 1000 and i < len(numbers):
    product *= numbers[i]
    i += 1
print(product)
```

In this example, the loop executes as long as the product of the numbers in the list is less than or equal to 1000, and there are still numbers left in the list to multiply. Inside the loop, the product is multiplied by the current number in the list, and the index i is incremented by 1. This continues until either the product exceeds 1000, or there are no more numbers left in the list.

- 3. When you want to repeatedly prompt the user for input until a certain condition is met:

In []:

```
# Prompt the user for a password until it meets the required complexity
criteria
password = ""
while len(password) < 8 or not any(char.isdigit() for char in password):
    password = input("Enter a password (must be at least 8 characters and
contain at least one digit): ")
print("Password set successfully!")
```

In this example, the loop executes as long as the user's input does not meet the required complexity criteria for a password (at least 8 characters long and containing at least one digit). Inside the loop, the user is prompted for input, and the input is checked to see if it meets the criteria. This continues until the user enters a password that meets the criteria.

Overall, while loops are useful when you want to repeatedly execute a block of code until a certain condition is met. However, you should be careful to avoid infinite loops (loops that never terminate), which can cause your program to hang or crash. Make sure that the condition in your while loop eventually becomes False, or include a mechanism for breaking out of the loop if necessary.

6. Use nested while loop to print 3 different pattern.

In [35]:

```
# Pattern 1
i = 1
while i <= 5:
    j = 1
    while j <= i:
        print(j, end=' ')
        j += 1
    print()
    i += 1
```

```
1
1 2
1 2 3
1 2 3 4
1 2 3 4 5
```

In [36]:

```
# Pattern 2
i = 1
while i <= 5:
    j = 1
    while j <= i:
        print(i, end=' ')
        j += 1
    print()
    i += 1
```

```
1
```

2 2
3 3 3
4 4 4 4
5 5 5 5 5

In [37]:

```
# Pattern 3
i = 5
while i >= 1:
    j = 1
    while j <= i:
        print(j, end=' ')
        j += 1
    print()
    i -= 1
```

1 2 3 4 5
1 2 3 4
1 2 3
1 2
1

7.Reverse a while loop to display numbers from 10 to 1

In [38]:

```
i = 10
while i >= 1:
    print(i)
    i -= 1
```

10
9
8
7
6
5
4
3
2
1