

Topics

Modules- 1.Random 2.DateTime 3.Math

RANDOM MODULE

1.Create a function that generates a random password of a specified length. The password should include uppercase letters, lowercase letters, digits, and special characters.

```
import random
import string

def generate_password(length):
    characters = string.ascii_letters + string.digits + string.punctuation
    password = ''.join(random.choice(characters) for _ in range(length))
    return password

password=generate_password(12)
print("Your password is:",password)
```

2.Write a function that simulates rolling a pair of six-sided dice 100 times. Count how many times the sum of the two dice is equal to 7.

```
import random

def count_dice_sum_seven(num_rolls=100):
    count_seven = 0

    for _ in range(num_rolls):
        # Roll two six-sided dice
        dice1 = random.randint(1, 6)
        dice2 = random.randint(1, 6)

        # Check if the sum is 7
        if dice1 + dice2 == 7:
            count_seven += 1

    return count_seven

# Simulate rolling dice 100 times and count how many times the sum is 7
num_sevens = count_dice_sum_seven()

print(f"Number of times the sum is 7: {num_sevens}")
```

➞ Number of times the sum is 7: 16

3.Create a program that picks 6 unique random numbers from 1 to 49, simulating a lottery draw

```
import random

def lottery_draw():
    lottery_numbers = []
    while len(lottery_numbers) < 6:
        number = random.randint(1, 49)
        if number not in lottery_numbers:
            lottery_numbers.append(number)

    return lottery_numbers

winning_numbers = lottery_draw()

print("Winning numbers:", winning_numbers)
```

➞ Winning numbers: [13, 5, 30, 3, 29, 1]

4.Write a program that randomly selects and displays a quote from a predefined list of quotes each time it is run.

```
import random
def rndom_quotes():
    quotes=["Empower the Youths",
            "Do Things Today,not tommorow",
            "Stay hungry, stay foolish. - Steve Jobs",
            "Your time is limited, so don't waste it living someone else's life. - Steve Jobs",
            "Life is what happens when you're busy making other plans. - John Lennon",
            "Strive not to be a success, but rather to be of value. - Albert Einstein",
            "Two things are infinite: the universe and human stupidity; and I'm not sure about the universe. - Albert Einstein",
            "The only source of knowledge is experience. - Albert Einstein",
            "Imagination is more important than knowledge. - Albert Einstein""Once you have decide to do no one can ruine it"]
    choose=random.choice(quotes)
    rndom_quotes()
    print("Random choice")
    print(choose)
```

DATETIME Module

1. Write a program that converts the current local time to UTC (Coordinated Universal Time). Display both the local time and the converted UTC time.

```
import datetime
import pytz

def convert_to_utc():
    local_time = datetime.datetime.now()
    local_timezone = pytz.timezone('UTC')
    local_time = local_timezone.localize(local_time)
    utc_time = local_time.astimezone(pytz.utc)

    local_time_str = local_time.strftime('%Y-%m-%d %H:%M:%S %Z%z')
    utc_time_str = utc_time.strftime('%Y-%m-%d %H:%M:%S UTC')

    print(f"Local Time: {local_time_str}")
    print(f"UTC Time: {utc_time_str}")

convert_to_utc()
```

```
➦ Local Time: 2024-06-20 11:48:34 UTC+0000
  UTC Time: 2024-06-20 11:48:34 UTC
```

2. Write a script that asks the user to input a date (in YYYY-MM-DD format) and prints the day of the week for that date.

```
import datetime

def get_day_of_week():
    value = input("Enter a date (YYYY-MM-DD): ")

    try:
        date_obj = datetime.datetime.strptime(value, '%Y-%m-%d')

        day_of_week = date_obj.strftime('%A')

        # Print the day of the week
        print(f"The day of the week for {value} is: {day_of_week}")

    except ValueError:
        print("Invalid date format")

get_day_of_week()
```

```
➦ Enter a date (YYYY-MM-DD): 2004-11-30
  The day of the week for 2004-11-30 is: Tuesday
```

3. Create a program that asks the user for their date of birth in YYYY-MM-DD format and calculates their current age in years, months, and days.

```
import datetime

def calculate_age():
    # Prompt user to input their date of birth
    dob_str = input("Enter your date of birth (YYYY-MM-DD): ")

    try:
        # Parse the input date string into a datetime object
        dob = datetime.datetime.strptime(dob_str, '%Y-%m-%d').date()

        # Get the current date
        current_date = datetime.date.today()

        # Calculate the difference between current date and date of birth
        age = current_date - dob

        # Calculate years, months, and days from the age difference
        years = age.days // 365
        months = (age.days % 365) // 30
        days = (age.days % 365) % 30

        # Print the age in years, months, and days
        print(f"Your age is: {years} years, {months} months, and {days} days.")

    except ValueError:
        print("Invalid date format. Please enter date in YYYY-MM-DD format.")

# Call the function to calculate age based on date of birth
calculate_age()
```

```
↩ Enter your date of birth (YYYY-MM-DD): 2004-11-30
Your age is: 19 years, 6 months, and 27 days.
```

MATH MODULES

1. Write a program that asks the user to input the radius of a circle. Calculate and print the area and circumference of the circle using the value of π from the math module

```
import math

def calculate():
    radius = float(input("Enter the radius of the circle: "))

    # Calculate area and circumference
    area = math.pi * radius**2
    circumference = 2 * math.pi * radius

    print(f"Radius of the circle: {radius}")
    print(f"Area of the circle: {area:.2f}")
    print(f"Circumference of the circle: {circumference:.2f}")

calculate()
```

```
↩ Enter the radius of the circle: 4
Radius of the circle: 4.0
Area of the circle: 50.27
Circumference of the circle: 25.13
```

2. Create a program that accepts the lengths of the two shorter sides of a right triangle. Calculate and print the length of the hypotenuse using the Pythagorean theorem.

```
import math

def calculate_hypotenuse():
    side1 = float(input("Enter the length of the first shorter side: "))
    side2 = float(input("Enter the length of the second shorter side: "))

    hypotenuse = math.sqrt(side1**2 + side2**2)

    # Print the calculated length of the hypotenuse
    print(f"The length of the hypotenuse is: {hypotenuse:.2f}")

calculate_hypotenuse()
```

↻ Enter the length of the first shorter side: 5
Enter the length of the second shorter side: 7
The length of the hypotenuse is: 8.60

3. Create a program that asks the user to input a number and a base. Calculate and print the logarithm of the number to the given base using the math module.

```
import math

def calculate_logarithm():
    number = float(input("Enter the number: "))
    base = float(input("Enter the base: "))

    logarithm = math.log(number, base)

    print(f"The logarithm of {number} to the base {base} is: {logarithm:.4f}")

calculate_logarithm()
```

↻ Enter the number: 1
Enter the base: 10
The logarithm of 1.0 to the base 10.0 is: 0.0000