Shahad Hamed Al Housni Shahadhamed1820@gmail.com

Problem 1 :

```
import re

def validate_username(username):
    if not username:
        return "Username should not be empty."
    if len(username) > 50:
        return "Username should not exceed 50 characters."
    return None

def validate_password(password):
    if len(password) < 8:
        return "Password must be at least 8 characters long."
    if not re.search(r"[!@#$%^&*(),.?\":{}|<>]", password):
        return "Password must contain at least one special symbol."
    if not re.search(r"\d", password):
        return "Password must contain at least one number."
    if not re.search(r"[A-Z]", password):
        return "Password must contain at least one uppercase letter."
    if not re.search(r"[a-z]", password):
        return "Password must contain at least one lowercase letter."
    return None

def validate_email(email):
    if "@" not in email:
        return "Email should contain '@' symbol."
    username_domain = email.split("@")
    if len(username_domain) != 2:
        return "Email should contain one '@' symbol."
```

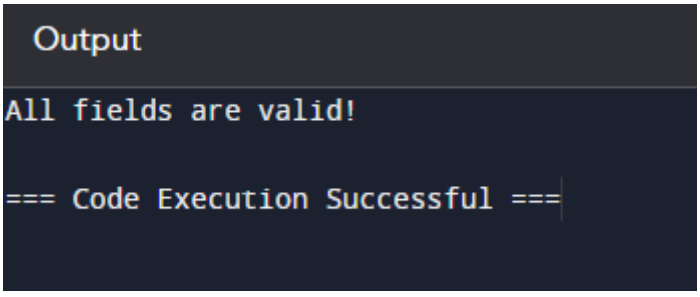
```
    username, domain = username_domain
    if not username.isalnum():
        return "Email should have alphanumeric characters before '@' symbol."
    if "." not in domain:
        return "Email should contain '.' after '@' symbol."
    domain_parts = domain.split(".")
    if any(not part.isalpha() for part in domain_parts):
        return "Email domain should contain only letters."
    return None

def main():
    username = "ShahadHamed"
    password = "Sh@96961121s"
    email = "shahadhamed1820@gmail.com"

    username_error = validate_username(username)
    password_error = validate_password(password)
    email_error = validate_email(email)

    if username_error:
        print(f"Invalid Username: {username_error}")
    elif password_error:
        print(f"Invalid Password: {password_error}")
    elif email_error:
        print(f"Invalid Email: {email_error}")
    else:
        print("All fields are valid!")

if __name__ == "__main__":
    main()
```



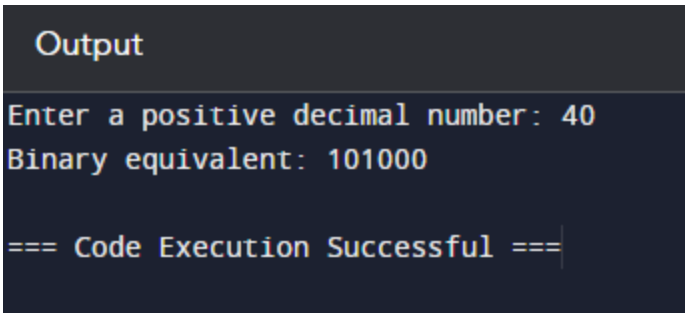
Shahad Hamed Al Housni Shahadhamed1820@gmail.com

Problem 2 :

```
def decimal_to_binary(n):
    if n == 0:
        return "0"
    binary = ""
    while n > 0:
        binary = str(n % 2) + binary
        n = n // 2
    return binary

def main():
    decimal_number = int(input("Enter a positive decimal number: "))
    if decimal_number < 0:
        print("Please enter a positive number.")
    else:
        print(f"Binary equivalent: {decimal_to_binary(decimal_number)}")

if __name__ == "__main__":
    main()
```



Shahad Hamed Al Housni Shahadhamed1820@gmail.com

Problem 3 :

```
def display_menu():
    print("1. Print a right-angled triangle")
    print("2. Print an equilateral triangle")
    print("3. Print an inverted right-angled triangle")
    print("4. Exit")

def right_angled_triangle(n):
    for i in range(1, n + 1):
        print('*' * i)

def equilateral_triangle(n):
    for i in range(1, n + 1):
        print(' ' * (n - i) + '*' * (2 * i - 1))

def inverted_right_angled_triangle(n):
    for i in range(n, 0, -1):
        print('*' * i)

def main():
    while True:
        display_menu()
        choice = int(input("Enter your choice (1-4): "))
        if choice == 4:
            print("Exiting...")
            break
        n = int(input("Enter the number of rows: "))
        if choice == 1:
            right_angled_triangle(n)
```

```
    elif choice == 2:
        equilateral_triangle(n)
    elif choice == 3:
        inverted_right_angled_triangle(n)
    else:
        print("Invalid choice! Please select a valid option.")

if __name__ == "__main__":
    main()
```

Output

```
1. Print a right-angled triangle
2. Print an equilateral triangle
3. Print an inverted right-angled triangle
4. Exit
Enter your choice (1-4): 1
Enter the number of rows: 4
*
**
***
****
1. Print a right-angled triangle
2. Print an equilateral triangle
3. Print an inverted right-angled triangle
4. Exit
Enter your choice (1-4): 2
Enter the number of rows: 3
  *
 ***
*****
1. Print a right-angled triangle
2. Print an equilateral triangle
3. Print an inverted right-angled triangle
4. Exit
Enter your choice (1-4): 3
Enter the number of rows: 3
***
**
*
1. Print a right-angled triangle
2. Print an equilateral triangle
3. Print an inverted right-angled triangle
4. Exit
Enter your choice (1-4): 4
Exiting...

=== Code Execution Successful ===
```

Shahad Hamed Al Housni Shahadhamed1820@gmail.com

Problem 4 :

```
def even_squares(lst):
    return [x**2 for x in lst if x % 2 == 0]

def slice_sublist(lst, start, end):
    return lst[start:end]

def main():
    lst = list(map(int, input("Enter a list of integers separated by spaces: ").split()))
    print(f"List of squares of even numbers: {even_squares(lst)}")
    start = int(input("Enter the start index for slicing: "))
    end = int(input("Enter the end index for slicing: "))
    print(f"Sliced sublist: {slice_sublist(lst, start, end)}")

if __name__ == "__main__":
    main()
```

Output

```
Enter a list of integers separated by spaces: -1 0 1 2 3 4
List of squares of even numbers: [0, 4, 16]
Enter the start index for slicing: 2
Enter the end index for slicing: 4
Sliced sublist: [1, 2]

=== Code Execution Successful ===
```