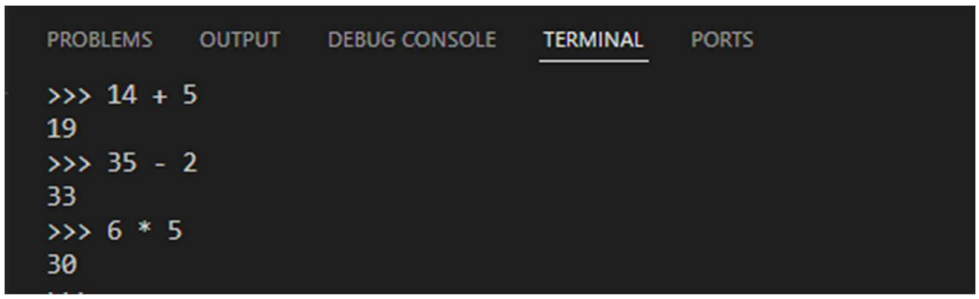
5a.

```
"""
Exp-5 Interactive Calculator
NAME: Hanif 01/231P044
"""
def parse_input(user_input):
    input_list = user_input.split()
    if len(input_list) != 3:
        raise Exception("Input must consist of three elements (e.g., '1 + 2')")
    n1, op, n2 = input_list
    try:
        n1 = float(n1)
        n2 = float(n2)
        # Check if both numbers are integers
        if not (n1.is_integer() and n2.is_integer()):
            raise Exception("Both operands must be integers")
        n1, n2 = int(n1), int(n2)  # Convert to integers
    except ValueError:
        raise Exception("Both inputs must be valid numbers")
    return n1, op, n2
def calculate(n1, op, n2):
    if op == '+':
        return n1 + n2
    elif op == '-':
        return n1 - n2
    elif op == '*':
        return n1 * n2
    elif op == '/':
```

```
        if n2 == 0:
            raise Exception("Division by zero is not allowed")
        return n1 / n2
    else:
        raise Exception(f"'{op}' is not a valid operator. Use +, -, *, or /")
while True:
    user_input = input('>>> ')
    if user_input.lower() == "quit":
        break
    try:
        n1, op, n2 = parse_input(user_input)
        result = calculate(n1, op, n2)
        print(result)
    except Exception as e:
        print(f"Error: {e}")
```

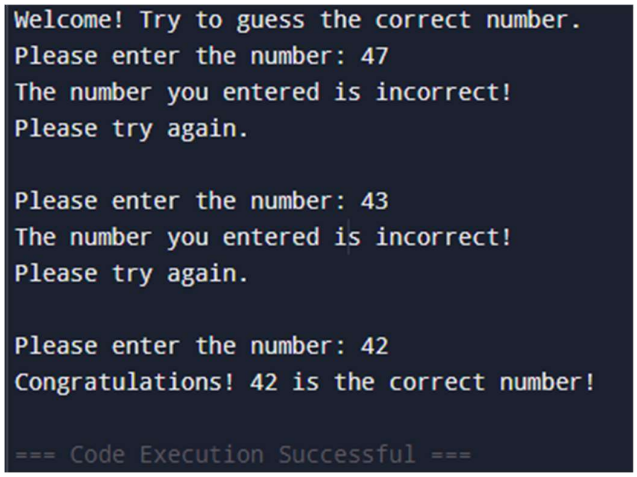
Output:



5b.

```
"""
Guess the number
NAME: QAYAM 231P038/02
"""
class IncorrectNumberError(Exception):
  def __init__(self, message="The number you entered is incorrect!"):
    self.message = message
    super().__init__(self.message)
def number_checker():
  correct_number = 12  # Define the correct number
  while True:
    try:
      user_input = int(input("Please enter the number: ")
      if user_input != correct_number:
        raise IncorrectNumberError
      else:
        print(f" Congratulations! {correct_number} is the correct number!")
        break  # Exit loop if correc
    except IncorrectNumberError as e:
      print(e)
      print(" Please try again.\n")
    except ValueError:
      print("Invalid input! Please enter a valid integer.\n")
if __name__ == "__main__":
  print("Welcome! Try to guess the correct number.")
  number_checker()
```

OUTPUT:

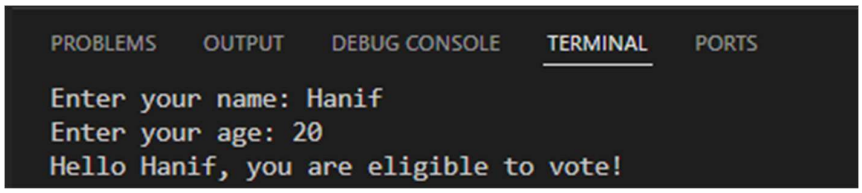


5c.

```
"""
Validate Name and Age
HANIF 231P044 / 01
"""
class InvalidAgeError(Exception):
    """Custom exception for invalid age input."""
    pas
class InvalidNameError(Exception):
    """Custom exception for invalid name input."""
    pas
def validate_name(name):
    if not name.replace(" ", "").isalpha():
        raise InvalidNameError("Invalid name! Name should contain only alphabets."
def validate_age(age):
    if age < 0:
        raise InvalidAgeError("Age cannot be negative!")
    elif age < 18:
        raise InvalidAgeError("You are not eligible to vote. Must be at least 18 years old.")
```

```
def main():
    try:
        name = input("Enter your name: ")
        validate_name(name)  # Validate name
        age = int(input("Enter your age: "))
        validate_age(age)  # Validate age
        print(f"Hello {name}, you are eligible to vote!")
    except InvalidNameError as e:
        print(e)
    except InvalidAgeError as e:
        print(e
    except ValueError:
        print("Invalid input! Age should be a valid number.")
if __name__ == "__main__":
    main()
```

OUTPUT:



5d.

```
"""
Program to Demonstrate User-Defined Exception
HANIF 231P044 / 01
"""
class InvalidMonthError(Exception):
    """Custom exception for invalid month number."""
    pass
```

```
def get_month_name(month_no):
    months = {
        1: "January", 2: "February", 3: "March", 4: "April",
        5: "May", 6: "June", 7: "July", 8: "August",
        9: "September", 10: "October", 11: "November", 12: "December"
    }
    if month_no not in months:
        raise InvalidMonthError("Invalid month number! Please enter a number between 1 and 12.")
    return months[month_no]
def main():
    try:
        month_no = int(input("Enter a month number (1-12): "))
        month_name = get_month_name(month_no)
        print(f"The month is: {month_name}")
    except InvalidMonthError as e:
        print(e
    except ValueError:
        print("Invalid input! Please enter a numeric value."
if __name__ == "__main__":
    main()
```

Output: