```python
class Student:
    def __init__(self, name, roll_no, branch):
        self.name = name
        self.roll_no = roll_no
        self.branch = branch

    def display_details(self):
        print("Student Name:", self.name)
        print("Roll Number:", self.roll_no)
        print("Branch:", self.branch)


# Object creation
student1 = Student("Vyshnavi", 101, "CSE")
student1.display_details()
```

```
Student Name: Vyshnavi
Roll Number: 101
Branch: CSE
```

```python
def print_multiples(num):
    for i in range(1, 11):
        print(num * i)

print_multiples(5)
```

```
5
10
15
20
25
30
35
40
45
50
```

```python
def print_multiples_while(num):
    i = 1
    while i <= 10:
        print(num * i)
        i += 1

print_multiples_while(5)
```

```
5
10
15
20
25
30
35
40
45
50
```

```python
def classify_age(age):
    if age < 13:
        return "Child"
    elif age < 20:
        return "Teenager"
    elif age < 60:
        return "Adult"
    else:
        return "Senior"

print(classify_age(25))
```

```
Adult
```

```python
def classify_age_dict(age):
    if age < 13:
        return "Child"
    if age < 20:
        return "Teenager"
```

```
    if age < 60:
        return "Adult"
    return "Senior"

print(classify_age_dict(65))
```

```
Senior
```

```
def sum_to_n(n):
    total = 0
    for i in range(1, n + 1):
        total += i
    return total

print(sum_to_n(10))
```

```
55
```

```
def sum_to_n_while(n):
    total = 0
    i = 1
    while i <= n:
        total += i
        i += 1
    return total

print(sum_to_n_while(10))
```

```
55
```

```
class BankAccount:
    def __init__(self, account_holder, balance=0):
        self.account_holder = account_holder
        self.balance = balance

    def deposit(self, amount):
        self.balance += amount
        print("Deposited:", amount)

    def withdraw(self, amount):
        if amount <= self.balance:
            self.balance -= amount
            print("Withdrawn:", amount)
        else:
            print("Insufficient balance")

    def check_balance(self):
        print("Current Balance:", self.balance)


# Object creation
account = BankAccount("Vyshnavi", 5000)
account.deposit(2000)
account.withdraw(1500)
account.check_balance()
```

```
Deposited: 2000
Withdrawn: 1500
Current Balance: 5500
```