

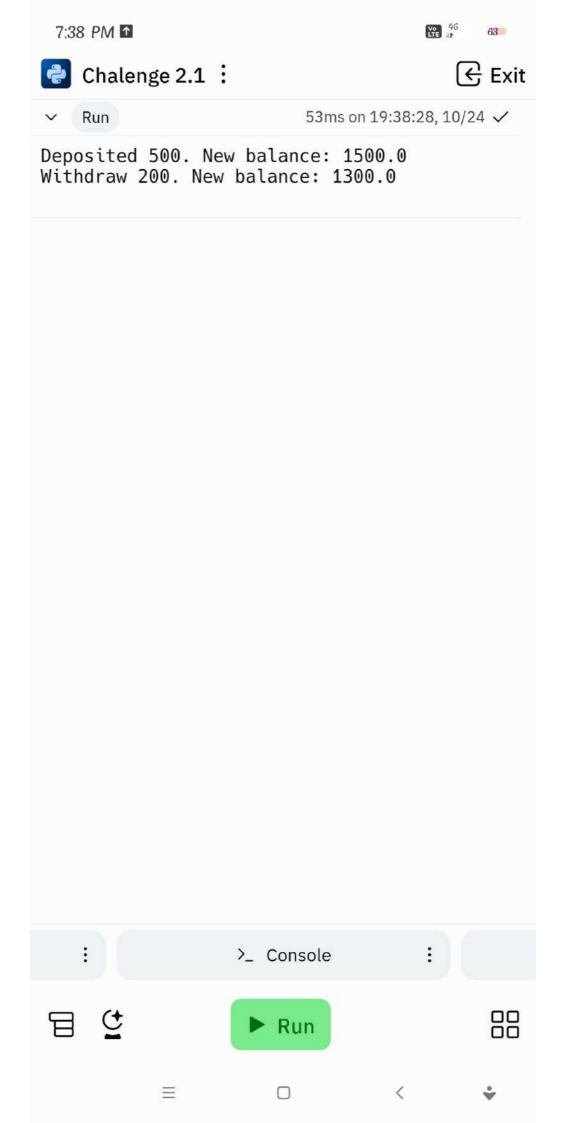
```
Vo 46+
 7:30 PM 1
                                             64
🔁 Chelange1. 2 ᠄
                                          € Exit
 1 \ def Fact_rec(n):
      if n==0 or n==1:
 2 ~
 3
         return 1
 4 ,
      else:
 5
        return n* Fact_rec(n-1)
 6
      number=2
 7
      res= Fact_rec(number)
      print ("the Factorial of
 8
    {}is{}.".format(number,res))
 9
10
                       Ln 1, Col 1 • Spaces: 2 History '5
                    襣 main.py
                                             Run
```

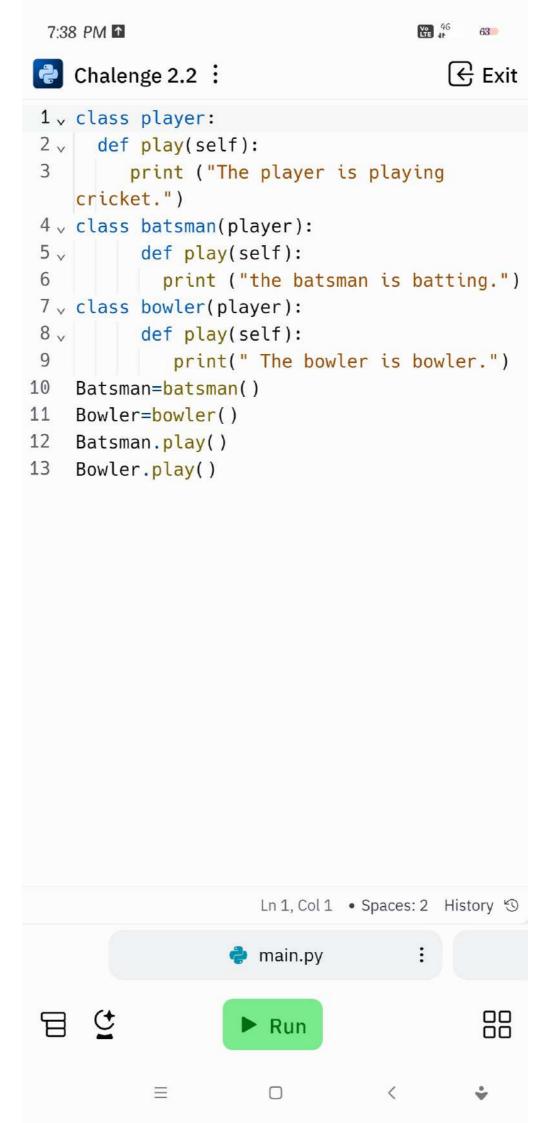
```
account number
             self.__account_holder_name =
4
    account_holder_name
5
             self.__account_balance =
    initial balance
6
7 ,
        def deposit(self, amount):
8 ~
             if amount > 0:
9
                 self.__account_balance +=
    amount
10
                 print('Deposited {}. New
    balance: {}'.format(amount,
    self.__account_balance))
11 ~
             else:
12
                 print('Invalid deposit
    amount. Please deposit a positive
    amount.')
13
14 \
        def withdraw(self, amount):
             if amount > 0 and amount <=
15 ~
    self.__account_balance:
                 self.__account_balance -=
16
    amount
17
                 print('Withdraw {}. New
                      Ln 1, Col 1 • Spaces: 2 History '5
                     main.py
                       Run
```

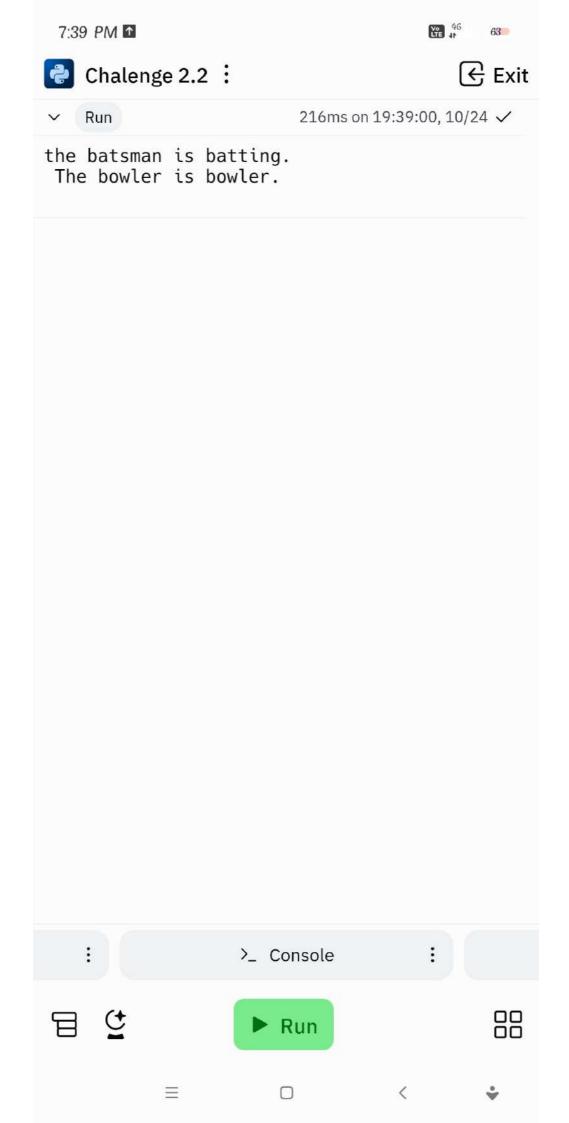
```
7:38 PM 1
                                      Vo 46
                                            63

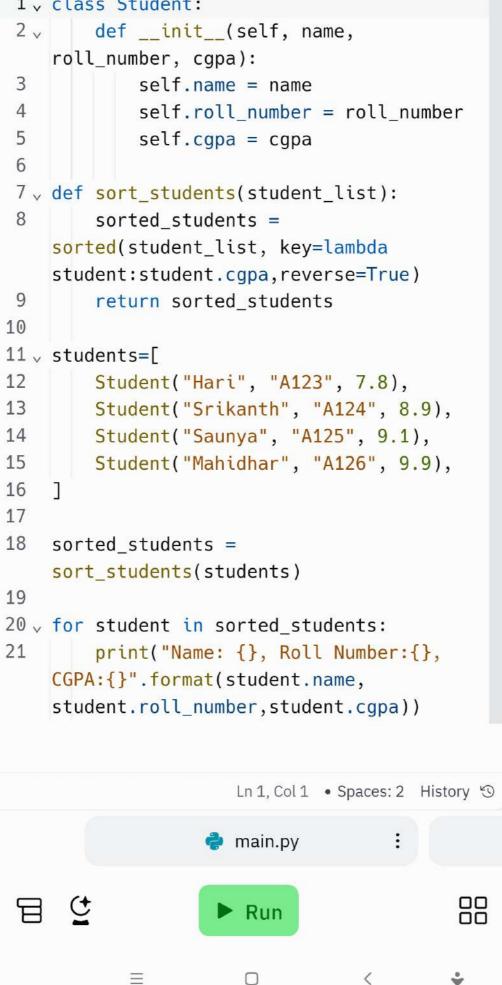
← Exit

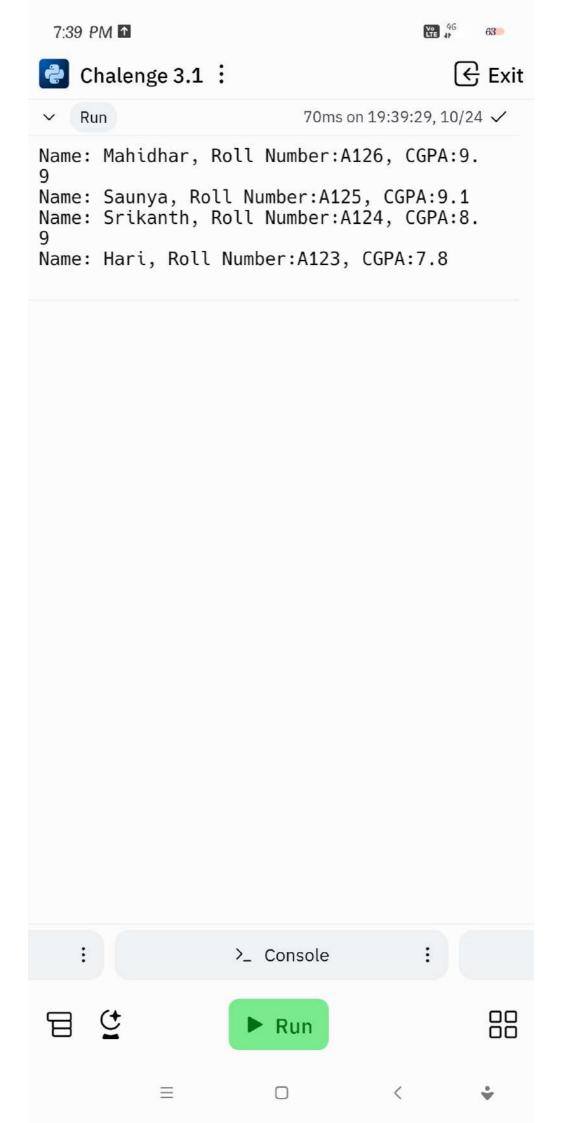
Chalenge 2.1
10
                 print('Deposited {}. New
    balance: {}'.format(amount,
    self.__account_balance))
11 \
            else:
                 print('Invalid deposit
12
    amount. Please deposit a positive
    amount.')
13
        def withdraw(self, amount):
14 \
15 .
             if amount > 0 and amount <=
    self.__account_balance:
16
                 self.__account_balance -=
    amount
                 print('Withdraw {}. New
17
    balance: {}'.format(amount,
    self.__account_balance))
18 ~
             else:
19
                 print('Invalid withdraw
    amount or insufficient balance.')
20
    # Example usage:
21
22
    account = BankAccount("12345", "John
    Doe", 1000.0)
23
    account.deposit(500)
24
    account.withdraw(200)
                      Ln 1, Col 1 • Spaces: 2 History '5
                     main.py
                       Run
```











```
7:40 PM 1
                                      Vo 46
                                            63

← Exit

Chalenge 3.2
 1 v def
    linear_search_product_list(productList,
    targetProduct):
 2
        indices = []
3 ,
        for index, product in
    enumerate(productList):
             if product == targetProduct:
 4 ,
 5
                 indices.append(index)
 6
        return indices
7
 8
    # Example usage:
    products = ["shoes", "boot", "loafes",
 9
    "shoes", "sandal", "shoes"]
    target = "shoes"
10
11
    target2 = "apple"
12
    result =
    linear_search_product_list(products,
    target)
    print(result)
13
                      Ln 1, Col 1 • Spaces: 2 History '5
                     main.py
                      Run
```

