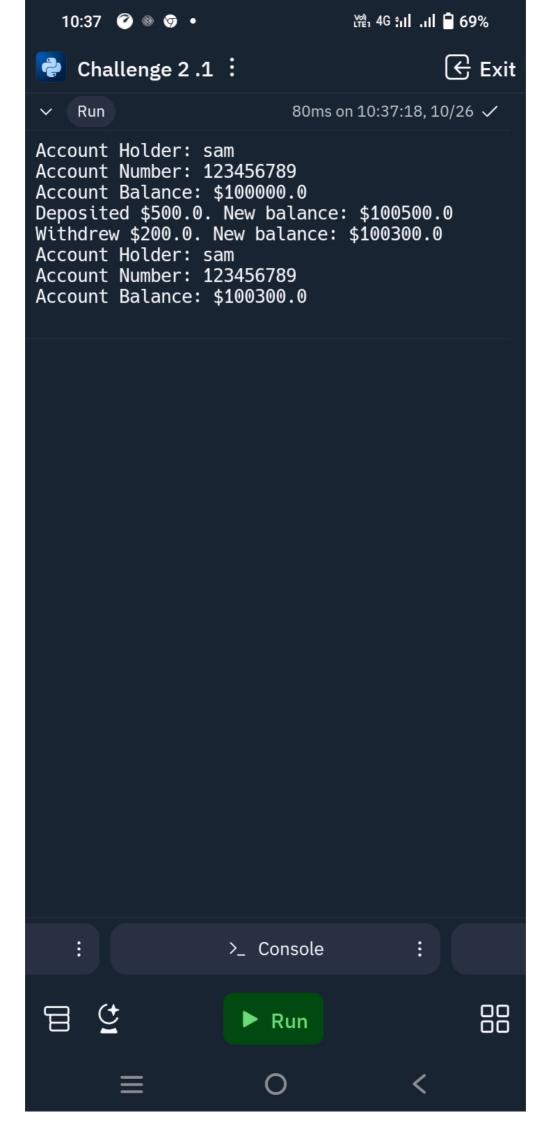
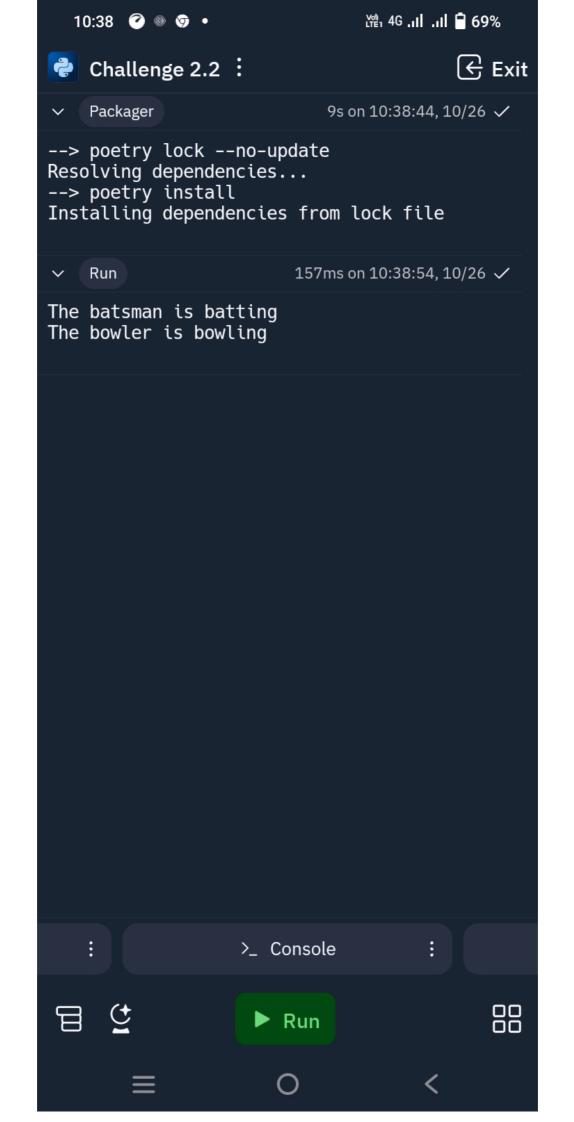
```
10:37 🕜 🌚 🐬 •
                               (약) 4G : ill .ill 🗖 69%
                                        ← Exit
    Challenge 2.1 :
 1 v class BankAccount:
2 \ def __init__(self, account_number,
    account holder name, initial balance):
             self.__account_number =
 3
    account number
4
             self.__account_holder_name =
    account_holder_name
             self.__account_balance =
 5
    initial balance
 6
       def deposit(self, amount):
8 ~
            if amount > 0:
9
                 self.__account_balance +=
    amount
10
                 print(f"Deposited
    ${amount}. New balance:
    ${self.__account_balance}")
11 🗸
             else:
12
                 print("Invalid deposit
    amount. Please enter a positive
    value.")
13
14 🗸
        def withdraw(self, amount):
            if amount > 0 and amount <=
15 <sub>~</sub>
    self.__account_balance:
16
                 self.__account_balance -=
    amount
17
                 print(f"Withdrew
    ¢{amount}
                New halance:
                      Ln 1, Col 1 • Spaces: 2 History 🔊
                   e main.py
                     Run
```

```
10:38 🕜 🕲 🦁 •
                               [변] 4G ::il .:il 🗎 69%
                                        ← Exit
    Challenge 2.1
    ${amount}. New balance:
    ${self. account balance}")
18 🗸
             else:
19
                 print("Invalid withdrawal
    amount or insufficient balance.")
20
21 \
        def display_balance(self):
             print(f"Account Holder:
22
    {self. account holder name}")
23
             print(f"Account Number:
    {self. account number}")
24
             print(f"Account Balance:
    ${self.__account_balance}")
25
26
27
28 \lor if __name__ == "__main__":
29
30
        my_account =
    BankAccount("123456789", "sam",
    100000.0)
31
32
33
        my_account.display_balance()
34
35
36
        my_account.deposit(500.0)
37
38
                      Ln 1, Col 1 • Spaces: 2 History 🔊
                   🍦 main.py
                      Run
```



```
10:38 🕜 🕲 🦁 •
                              (월, 4G : ill . ill 🗎 69%
                                       ← Exit
challenge 2.2
 1 v class Player:
2 \ def play(self):
             print("The player is playing
 3
    cricket")
 4
5 v class Batsman(Player):
6 def play(self):
7
            print("The batsman is batting")
8
9 v class Bowler(Player):
10 def play(self):
11
            print("The bowler is bowling")
12
13
14
    batsman = Batsman()
    bowler = Bowler()
15
16
17
18
    batsman.play()
    bowler.play()
19
                     Ln 1, Col 1 • Spaces: 2 History 🔊
                  e main.py
 冒
                     Run
```



```
(학 4G : ill .ill 🗎 69%
  10:39
            9
                                         Exit
🕏 Challenge 3.1 :
1 v def linear_search_product(product_list,
   target_product):
        indices = []
2
        for i, product in
   enumerate(product_list):
            if product == target_product:
4 🗸
                 indices.append(i)
6
        return indices
                      Ln 1, Col 1 • Spaces: 2 History 🔊
                   e main.py
                     Run
```

```
10:39 🕜 🚳 🕲 🔹
                              (약) 4G .ill .ill 🗎 69%
                                       ← Exit
🕏 Challenge 3.2 :
 1 v class Student:
2 def init (self, name,
    roll number, cqpa):
 3
             self.name = name
             self.roll number = roll number
 4
5
            self.cgpa = cgpa
 6
7 \ def sort_students(student_list):
        sorted students =
    sorted(student_list, key=lambda
    student: student.cgpa, reverse=True)
9
        return sorted_students
10
11
12 \vee students = [
13
        Student("Alice", "A123", 3.9),
        Student("Bob", "B456", 3.7),
14
        Student("Charlie", "C789", 3.5),
15
16
        Student("David", "D234", 3.8),
17
    1
18
19
    sorted students =
    sort students(students)
20
21 v for student in sorted students:
22
        print(f"Name: {student.name}, Roll
    Number: {student.roll_number}, CGPA:
    {student.cgpa}")
                     Ln 1, Col 1 • Spaces: 2 History 🔊
                  e main.py
                     Run
```

