



## Challenge 2.1 :



Exit

```
1 v class BankAccount:
2 v     def __init__(self, account_number,
   account_holder_name, initial_balance):
3         self.__account_number =
   account_number
4         self.__account_holder_name =
   account_holder_name
5         self.__account_balance =
   initial_balance
6
7 v     def deposit(self, amount):
8 v         if amount > 0:
9             self.__account_balance +=
   amount
10             print(f"Deposited
   ${amount}. New balance:
   ${self.__account_balance}")
11 v         else:
12             print("Invalid deposit
   amount. Please enter a positive
   value.")
13
14 v     def withdraw(self, amount):
15 v         if amount > 0 and amount <=
   self.__account_balance:
16             self.__account_balance -=
   amount
17             print(f"Withdrew
   ${amount}. New balance:
   ${self.__account_balance}")
```

Ln 1, Col 1 • Spaces: 2 History



main.py



Run





## Challenge 2.1 :



Exit

```
{amount}. New balance:
${self.__account_balance}")
18     else:
19         print("Invalid withdrawal
amount or insufficient balance.")
20
21     def display_balance(self):
22         print(f"Account Holder:
{self.__account_holder_name}")
23         print(f"Account Number:
{self.__account_number}")
24         print(f"Account Balance:
${self.__account_balance}")
25
26
27 # Testing the BankAccount class
28 if __name__ == "__main__":
29     # Create an instance of BankAccount
30     my_account =
BankAccount("123456789", "sam",
100000.0)
31
32     # Display initial balance
33     my_account.display_balance()
34
35     # Deposit money
36     my_account.deposit(500.0)
37
38     # Withdraw money
```

Ln 1, Col 1 • Spaces: 2 History



main.py



Run





## Challenge 2.1 :



Exit



Run

80ms on 10:37:18, 10/26 ✓

```
Account Holder: sam
Account Number: 123456789
Account Balance: $100000.0
Deposited $500.0. New balance: $100500.0
Withdrew $200.0. New balance: $100300.0
Account Holder: sam
Account Number: 123456789
Account Balance: $100300.0
```



&gt;\_ Console



Run





## Challenge 2.2 :

Exit

```
1 v class Player:
2 v     def play(self):
3         print("The player is playing
         cricket")
4
5 v class Batsman(Player):
6 v     def play(self):
7         print("The batsman is batting")
8
9 v class Bowler(Player):
10 v     def play(self):
11         print("The bowler is bowling")
12
13 # Creating objects of Batsman and
    Bowler classes
14 batsman = Batsman()
15 bowler = Bowler()
16
17 # Calling the play() method for each
    object
18 batsman.play()
19 bowler.play()
```

Ln 1, Col 1 • Spaces: 2 History



main.py



Run





## Challenge 2.2 :

Exit



Packager

9s on 10:38:44, 10/26 ✓

```
--> poetry lock --no-update  
Resolving dependencies...  
--> poetry install  
Installing dependencies from lock file
```



Run

157ms on 10:38:54, 10/26 ✓

```
The batsman is batting  
The bowler is bowling
```



&gt;\_ Console



Run






## Challenge 3.1 :



Exit

```
1 v def linear_search_product(product_list,
    target_product):
2     indices = []
3 v     for i, product in
        enumerate(product_list):
4 v         if product == target_product:
5             indices.append(i)
6     return indices
```

Ln 1, Col 1 • Spaces: 2 History 

main.py



Run





## Challenge 3.2 :



Exit

```
1 v class Student:
2 v     def __init__(self, name,
    roll_number, cgpa):
3         self.name = name
4         self.roll_number = roll_number
5         self.cgpa = cgpa
6
7 v def sort_students(student_list):
8     sorted_students =
    sorted(student_list, key=lambda
    student: student.cgpa, reverse=True)
9     return sorted_students
10
11 # Example usage:
12 v students = [
13     Student("Alice", "A123", 3.9),
14     Student("Bob", "B456", 3.7),
15     Student("Charlie", "C789", 3.5),
16     Student("David", "D234", 3.8),
17 ]
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



main.py



Run





## Challenge 3.2 :

Exit



Packager

12s on 10:40:02, 10/26 ✓

```
--> poetry lock --no-update  
Resolving dependencies...
```

```
Writing lock file
```

```
--> poetry install
```

```
Installing dependencies from lock file
```



Run

116ms on 10:40:15, 10/26 ✓

```
Name: Alice, Roll Number: A123, CGPA: 3.9  
Name: David, Roll Number: D234, CGPA: 3.8  
Name: Bob, Roll Number: B456, CGPA: 3.7  
Name: Charlie, Roll Number: C789, CGPA: 3.  
5
```



&gt;\_ Console



Run

