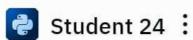




```
1 v class bankaccount:
2 def __init__(self,
    account_number,
    account_holder_name,
    initial balance=0.0):
        self.__account_number =
3
    account number
        self.__account_holder_name =
4
    account_holder_name
        self. account balance =
5
    initial balance
6
7 _
     def deposit(self, amount):
8 ,
        if amount > 0:
          self. account balance
9
    +=amount
10
          print("Deposited ${}. New
    balance: ${}". format(amount,
    self. account balance))
11 ~
        else:
12
          print("Invalid deposit
    amount,")
    def withdraw(self, amount):
13 ~
14 \
        if amount > 0 and amount <=
    self.__account_balance:
15
          self.__account_balance -=
    amount
          print("withdrew ${}. New
16
                           Ln 1, Col 1 History 5
                 main.py
```

Run





```
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 \ def sort_students(student_list):
7
    sorted_students=sorted(student_list,
8
         key=lambda
    student:student.cgpa,
9
                            reverse=True)
10
       return sorted students
11 v students=
    [Student("Hari", "A123", 7.8),
12
    Student("Srikanth", "A124", 8.9),
13
    Student("Saumya", "A125", 9.1),
14
    Student("Mahidhar", "A126", 9.9),
15
16
    sorted students =
    sort_students(students)
17 v for student in sorted_students:
      print("name: {},roll number:
18
    {},CGPA:
                            Ln 1, Col 1 History 5
                 main.py
```

Run



**€** Exit

```
1 v def fact_rec(n):
2 \sqrt{\text{if n}} = 0 \text{ or n} = 1:
3
       return 1
4 ,
    else:
5
        return n*fact_rec(n-1)
   number = 6
6
   res = fact_rec(number)
7
   print("the factorial of {} is
   {}.".format(number, res))
                              Ln 1, Col 1 History '9
                   e main.py
```

Run



```
l Leap24
                                   € Exit
1 v def cheakleap(year):
2
     if((year % 400==0)or
3
        (year % 100!=0)and
4 ,
        (year % 4 ==0)):
5
       print("given year a leap year")
6 ,
     else:
7
       print("given year is not a leap
   year")
   year = int(input("enter the number"))
8
   cheakleap(year)
9
```

Ln 1, Col 1 History 'S



e main.py













## **?** Player 24

```
Exit
```

```
1 v class Player:
2 \ def play(self):
3
        print("The player is playing
    cricket.")
4 v class Batsman(Player):
5 def play(self):
       print("The batsman is batting.")
7 v class Bowler(Player):
8 \ def play(self):
9
        print("The bowler is bowling.")
   batsman = Batsman()
10
11 bowler = Bowler()
12 batsman.play()
13 bowler.play()
```

Ln 1, Col 1 History 'S



e main.py









