



```
1 ▾ def fact_rec(n):  
2 ▾     if n==0 or n==1:  
3         return 1  
4 ▾     else:  
5         return n*fact_rec(n-1)  
6  
7     number=2  
8     res = fact_rec(number)  
9  
10    print("The factorial of{} is  
    {}".format(number,res,))  
11
```

Ln 1, Col 1 History ↺



main.py



Run





```

1  #Leap year|
2  year=int(input("enter year to be
   checked:"))
3  ✓ if year%4==0:
4  ✓     if year%100==0:
5  ✓         if year%400==0:
6             print("the year is a leap
   year!")
7  ✓     else:
8             print("the year is not a
   leap year!")
9  ✓ else:
10         print("the year is a leap
   year!")
11 ✓ else:
12         print("the year is not a leap
   year!")
13

```

🔴 main.py ⋮



■ Stop





```

1 v class BankAccount:
2
3 v     def __init__(self,
4         account_number,
5         account_holder_name,
6         initial_balance=0):
7
8         self.__account_number =
9         account_number
10
11         self.__account_holder_name =
12         account_holder_name
13
14         self.__account_balance =
15         initial_balance
16
17     def deposit(self, amount):
18
19         if amount > 0:
20
21             self.__account_balance += amount

```



main.py



Run





```

1 ▾ def
    linear_search_product(products,
    target_product):
2     indices = []
3 ▾     for index, product in
    enumerate(products):
4 ▾         if product==target_product:
5             indices.append(index)
6     return indices
7
8 #Example usage:
9
10 # Sample list of products
11 product_list =
    ["apple","banana", "orange",
    "apple", "grape", "apple"]
12
13 # Target product to search for
14 target_product = "apple"
15
16 # Call the function
17 result =
    linear_search_product(product_lis

```

Ln 1, Col 1 History ↺



main.py



Run





```
1 ▾ def sort_students(student_list):
2     sorted_students =
sorted(student_list, key=lambda
student: student.cgpa,
reverse=True)
3     return sorted_students
4
5 ▾ class Student:
6 ▾     def __init__(self, name,
roll_number, cgpa):
7         self.name = name
8         self.roll_number =
roll_number
9         self.cgpa = cgpa
10
11 # Test with different input
lists of students
12 ▾ students_list = [
13     Student("John", "2021001", 3.9),
14     Student("Jane", "2021002", 3.7),
15     Student("Alice", "2021003", 3.8),
```

Ln 1, Col 1 History



main.py



Run





```

1 ▾ class Player:
2
3 ▾     def play(self):
4
5         print("The player is
playing cricket.")
6
7
8
9
10
11 ▾ class Batsman(Player):
12
13 ▾     def play(self):
14
15         print("The batsman is
batting.")
16
17
18
19
20
21 ▾ class Bowler(Player):

```

Ln 1, Col 1 History ↺



main.py



Run





Cloud

92% COMPLETE



Next Lesson

Fundamentals of Cloud 1

Unit 1 - Fundamentals of Python



✓ 61 / 61 complete

Unit 2 - Object-Oriented Programming (OOP)



✓ 35 / 35 complete

Unit 3 - Data Structures and Manipulation



✓ 48 / 48 complete