Home Exercise 1

```
Name    : Jayed hoshen
ID      : 2204076
Section : B1
```

String Operations

Given a string "Hello World", reverse the string using slicing.

```
givenString = "Hello World"
print(givenString[::-1])
```

```
dlroW olleH
```

String Slicing

Given a string, extract a substring using slicing. For example, extract the word "quick" from the sentence, "The quick brown fox jumps over the lazy dog"

```
name = "The quick brown fox jumps over the lazy dog"
sz = name.find("quick")
print(name[sz : sz+5])
```

```
quick
```

String Split with a Specific Delimiter

Split a date string into day, month, and year. (e,g., date ="10-18-2024')

```
date = "10-18-2024"
date.split('-')
```

```
['10', '18', '2024']
```

Regular Expressions

Checks if a given email address is valid (contains "@" and ".com" or ".org").

```python
import re
mail = "jayedkn64@gmail.com"

if re.match(r"[^@]+@[^@]+\.(com/org)", mail) :
    print("Valid email")
else :
    print("Imvalid email")
```

```
Imvalid email
```

String Replace

Write a program that replaces all occurrences of the word "apple" with "orange" in a given string, "I like apple. Apple is sweet."

```python
import re
str = "I like apple. Apple is sweet."
pattern = r"apple"
replacement = "orange"
new_string = re.sub(pattern, replacement, str, flags=re.IGNORECASE
print(new_string)
```

```
I like orange. orange is sweet.
```

Dictionaries

Write a program that stores the names and ages of 3 people in a dictionary and prints the name of the person with the highest age.

```python
info = {"person1":22,"person2": 26, "person3":30}
max_age = max(info.values())
for name, age in info.items():
  if age == max_age:
    print(f"Name : '{name}'\nAge : {age}")
```

```
Name : 'person3'
Age : 30
```

Simple Student Attendance System

Problem:

Create a system that tracks attendance using a dictionary. The dictionary should store student names as keys and a list of strings as values, where each string is the date of their attendance.

Perform the following tasks:

- Add a new attendance record for a student.
- Remove a student from the system.
- Print the attendance of all students.

Sample Input:

attendance = { "Alice": ["2023-10-01", "2023-10-02"], "Bob": ["2023-10-01"] }

- Add: Attendance for "Alice" on "2023-10-03"
- Remove: "Bob"

```
attendance = { "Alice": ["2023-10-01", "2023-10-02"], "Bob": ["202

attendance["Alice"].append("2023-10-03")
attendance["Alice"].append("2023-10-04")
attendance["person1"]=["2023-10-03"]
attendance["person1"].append("2023-10-04")

del attendance["Bob"]
print(attendance)
```

```
{'Alice': ['2023-10-01', '2023-10-02', '2023-10-03', '2023-10-04'], 'person1': ['2023
```

Simple Phone Book Directory

Problem:

Create a phone book using a dictionary where names are the keys and phone numbers are the values. Perform the following tasks:

- Add a new contact.
- Remove a contact.
- Update a contact's phone number.
- Print the entire phone book.

Sample Input:

phone_book = {"John": "123-4567", "Jane": "987-6543"}

- Add: "Mike" with phone number "555-1212"
- Remove: "Jane"
- Update: "John" to "111-2222"

```
Phone_book = {"John": "123-4567", "Jane": "987-6543", "jayed":"222

```

```
Phone_book["Mike"] = "555-1212"
del Phone_book["Jane"]
Phone_book["John"] = "111-2222"

print(Phone_book)
```

```
{'John': '111-2222', 'jayed': '222-7809', 'Mike': '555-1212'}
```