**📝 Day 10 – Home Assignments: String Operations (Indexing, Slicing, Methods)**

**🔹 Part A: Indexing & Slicing**

1. **Basic Indexing**  
   Given: text = "PythonProgramming"
   * Print the **first**, **fifth**, and **last** character using indexing.
2. **Basic Slicing**  
   From the same string:
   * Print the substring "Python"
   * Print the substring "Programming"
   * Print every second character (::2)
   * Reverse the string using slicing
3. **Custom Substring**  
   Input a string from the user and print:
   * First 3 characters
   * Last 3 characters
   * Middle 5 characters

**🔹 Part B: String Methods Practice**

1. **split() Method**  
   Input: "apple,banana,cherry"
   * Split the string by comma and print the list.
2. **join() Method**  
   Given a list: ["red", "green", "blue"]
   * Join the list using - as separator: "red-green-blue"
3. **replace() Method**  
   Input: "The sky is blue"
   * Replace "blue" with "clear"
4. **lower() and upper()**
   * Convert "Python Is FUN" to all lowercase and all uppercase.
5. **count() and find()**  
   For the string "banana":
   * Count how many times "a" appears
   * Find the first occurrence of "n"

**🔹 Part C: Logical String Tasks**

1. **Vowel Counter**
   * Input a sentence and count how many vowels are present.
2. **Palindrome Checker**

* Check if a string is the same when reversed (e.g., "madam", "racecar")

1. **Word Count**

* Input a sentence and print total number of words.

1. **Character Frequency Dictionary**

* Input a string and create a dictionary of character counts.  
  Example: "book" → {'b':1, 'o':2, 'k':1}

**🔹 Part D: Real-World Tasks**

1. **Email Validator (Basic)**

* Input a string and check if it contains '@' and ends with ".com"

1. **Censor Bad Words**

* Input a sentence and replace "bad" and "ugly" with "\*\*\*"

1. **Format Name**

* Input a name like "john DOE" and convert it to "John Doe" (use title())

1. **Find the Longest Word**

* Input a sentence
* Find and print the **longest word** and its length

1. **Initials Generator**

* Input: "John Ronald Reuel Tolkien"
* Output: "J.R.R.T."

1. **URL Parser**

* Input a URL like "https://www.example.com/home"
* Extract and print:
  + Domain: example.com
  + Path: /home