**📝 Day 9 – Home Assignments: Dictionaries – Key-Value Pairs & Real-World Use**

**🔹 Part A: Dictionary Basics**

1. **Create a Dictionary**  
   Create a dictionary named student with keys:
   * "name" → "Alice"
   * "age" → 20
   * "course" → "Python"
   * Print the dictionary.
2. **Access Values**
   * Print the value of "name" from the student dictionary.
3. **Modify and Add**
   * Change "age" to 21
   * Add a new key "city" with value "Mumbai"
4. **Delete Key**
   * Remove the "course" key from the dictionary using del.

**🔹 Part B: Dictionary Operations**

1. **Loop Through Dictionary**  
   Write a loop to print each key and its value in the student dictionary.
2. **Check Key Existence**
   * Use in keyword to check if "name" exists in student.
3. **Get with Default**
   * Use get() to retrieve "grade" from the dictionary.  
     Provide default value "Not Assigned".

**🔹 Part C: Real-World Dictionary Use Cases**

1. **Contact Book**
   * Create a dictionary contacts with 3 people’s names as keys and their phone numbers as values.
   * Add a new contact.
   * Update one contact’s number.
   * Delete one contact.
2. **Inventory System**
   * Create a dictionary inventory where keys are product names ("pen", "notebook", "eraser") and values are quantities.
   * Write a function update\_stock(item, quantity) to increase stock.
   * If item is not in inventory, add it.
3. **Student Marks**

* Create a dictionary marks with subjects as keys and marks out of 100 as values.
* Calculate total marks and average.
* Print a message based on average:
  + >90: Excellent
  + 70–90: Good
  + <70: Needs Improvement

**🔹 Part D: Advanced Applications**

1. **Word Frequency Counter**

* Input a sentence from the user.
* Count how many times each word occurs using a dictionary.

1. **Character Frequency**

* Input a string.
* Count how many times each character occurs.

1. **Grades Lookup**

* Create a dictionary where keys are roll numbers and values are student names.
* Ask user for a roll number and print the corresponding name.

1. **Nested Dictionary**

* Create a nested dictionary:

student = {

"name": "Ravi",

"marks": {

"math": 88,

"science": 92

}

}

* Access "science" marks from the nested dictionary.

1. **Dictionary from Two Lists**

* Given:

keys = ["name", "age", "city"]

values = ["Sam", 30, "Delhi"]

* Combine into a dictionary using a loop or zip().