Python Masterclass-3 Assignment Answers

1. Dictionary Quest-

Task:

You are on a quest to collect various items in different locations. Your task is to create a dictionary to keep track of the items you find in each location. Write a Python program that allows you to add items to the dictionary, remove items, and check if a specific item exists in a location.

Answer:

```
# Create an empty dictionary to represent the player's inventory

inventory = {}


# Function to check if a specific item exists in any of the locations in the
inventory

def check_item_exists(item):

    for location, items in inventory.items():

        if item in items:

            return True

    return False


# Prompt the user to enter a location and an item found in that location

def add_item():

    location = input("Enter a location: ")

    item = input("Enter an item found in that location: ")


# Add the item to the inventory dictionary with the location as the key and
the item as the value

    if location in inventory:

        inventory[location].append(item)

    else:

        inventory[location] = [item]


# Allow the user to remove an item from a specific location

def remove_item():

    location = input("Enter the location to remove an item from: ")


# Check if the location exists in the inventory

    if location in inventory:

        item = input("Enter the item to remove: ")

# Check if the item exists in the specified location

        if item in inventory[location]:

            inventory[location].remove(item)

            print("Item removed successfully!")
```

Python Masterclass-3 Assignment Answers

```
        else:
            print("Item not found in the specified location.")
    else:
        print("Location not found in the inventory.")


# Display the current inventory
def display_inventory():
    print("Current Inventory:")
    for location, items in inventory.items():
        print(location + ": " + ", ".join(items))
# Main program loop
while True:
    print("\n-- Inventory Management --")
    print("1. Add an item to the inventory")
    print("2. Remove an item from the inventory")
    print("3. Check if an item exists in the inventory")
    print("4. Display current inventory")
    print("5. Quit")
    choice = input("Enter your choice (1-5): ")
    if choice == '1':
        add_item()
        display_inventory()
    elif choice == '2':
        remove_item()
        display_inventory()
    elif choice == '3':
        item = input("Enter the item to check: ")
        exists = check_item_exists(item)
        if exists:
            print("The item exists in the inventory.")
        else:
            print("The item does not exist in the inventory.")
    elif choice == '4':
        display_inventory()
    elif choice == '5':
        print("Goodbye!")
        break
    else:
        print("Invalid choice. Please try again.")
```

Python Masterclass-3 Assignment Answers

2. Set Collector-

Task:

You are a collector of unique items. Your task is to create sets of items based on their categories and perform operations on those sets. Write a Python program that allows you to create sets, combine sets, find common elements, and check if one set is a proper subset of another set.

Answer:

```
# Prompt the user to enter items and their categories
def prompt_items():
    items = {}
    while True:
        category = input("Enter the category of the item (or 'q' to quit): ")
        if category == 'q':
            break
        item = input("Enter an item in that category: ")
        if category in items:
            items[category].add(item)
        else:
            items[category] = {item}
    return items
# Combine two sets into a single set
def combine_sets(set1, set2):
    return set1.union(set2)
# Find and display the common elements between two sets
def find_common_elements(set1, set2):
    common_elements = set1.intersection(set2)
    print("Common elements:", common_elements)
# Check if one set is a proper subset of another set
def check_proper_subset(set1, set2):
    if set1.issubset(set2) and set1 != set2:
        print("Set 1 is a proper subset of Set 2.")
    else:
        print("Set 1 is not a proper subset of Set 2.")
# Main program loop
while True:
    print("\n-- Set Collector --")
    print("1. Enter items and their categories")
    print("2. Combine two sets into a single set")
    print("3. Find common elements between two sets")
    print("4. Check if one set is a proper subset of another set")
```

Python Masterclass-3 Assignment Answers

```
    print("5. Quit")
    choice = input("Enter your choice (1-5): ")
    if choice == '1':
        items = prompt_items()
        print("\nSets created:")
        for category, item_set in items.items():
            print(category + ": " + str(item_set))
    elif choice == '2':
        set1_category = input("Enter the category of the first set: ")
        set2_category = input("Enter the category of the second set: ")
# Check if the entered categories exist in the items dictionary
        if set1_category in items and set2_category in items:
            set1 = items[set1_category]
            set2 = items[set2_category]
            combined_set = combine_sets(set1, set2)
            print("Combined set:", combined_set)
        else:
            print("Invalid categories. Please try again.")
    elif choice == '3':
        set1_category = input("Enter the category of the first set: ")
        set2_category = input("Enter the category of the second set: ")
# Check if the entered categories exist in the items dictionary
        if set1_category in items and set2_category in items:
            set1 = items[set1_category]
            set2 = items[set2_category]
            find_common_elements(set1, set2)
        else:
            print("Invalid categories. Please try again.")
    elif choice == '4':
        set1_category = input("Enter the category of the first set: ")
        set2_category = input("Enter the category of the second set: ")
```

Python Masterclass-3 Assignment Answers

3. Number Challenge-

Task:

You have encountered a number challenge! Your task is to solve different number-related puzzles using control flow statements. Write a Python program that prompts the user to enter a number and perform operations to determine properties of the number.

Answer:

```
# Prompt the user to enter a number
number = float(input("Enter a number: "))

# Check if the number is positive, negative, or zero, and display the result
if number > 0:
    print("The number is positive.")
elif number < 0:
    print("The number is negative.")
else:
    print("The number is zero.")

# Check if the number is even or odd, and display the result
if number % 2 == 0:
    print("The number is even.")
else:
    print("The number is odd.")

# Prompt the user to enter a divisor
divisor = float(input("Enter a divisor: "))

# Check if the number is a multiple of the divisor, and display the result
if number % divisor == 0:
    print("The number is a multiple of the divisor.")
else:
    print("The number is not a multiple of the divisor.")

# Provide feedback to the user
if number > 0 and number % 2 == 0:
    print("Great job! You entered a positive even number.")
elif number < 0 and number % 2 != 0:
    print("Well done! You entered a negative odd number.")
else:
    print("Keep practicing! Challenge yourself with more number puzzles.")
```