

# ICP1 REPORT

```

▶ # accept a string and delete atleast 2 characters,reverse the resultant string
input_string = input("Enter a string: ")
char_list = list(input_string)
if len(char_list) < 2:
    print("The string is too short to delete two characters.")
else:
    del char_list[5]
    del char_list[4]
    resultant_string = ''.join(char_list)
    reversed_string = resultant_string[::-1]
    print("Reversed resultant string:", reversed_string)

```

```

⇒ Enter a string: sachin
Reversed resultant string: hcas

```

```

▶ # Take two numbers as input from the user for arithmetic operations
num1 = float(input("Enter the first number: "))
num2 = float(input("Enter the second number: "))
addition = num1 + num2
subtraction = num1 - num2
multiplication = num1 * num2
try:
    division = num1 / num2
except ZeroDivisionError:
    division = "undefined (cannot divide by zero)"
print(f"Addition: {num1} + {num2} = {addition}")
print(f"Subtraction: {num1} - {num2} = {subtraction}")
print(f"Multiplication: {num1} * {num2} = {multiplication}")
print(f"Division: {num1} / {num2} = {division}")

```

```

⇒ Enter the first number: 20
Enter the second number: 10
Addition: 20.0 + 10.0 = 30.0
Subtraction: 20.0 - 10.0 = 10.0
Multiplication: 20.0 * 10.0 = 200.0
Division: 20.0 / 10.0 = 2.0

```

```

▶ # accept a sentence python and replace with pythons
sentence = input("Enter sentence: ")
modified_sentence = sentence.replace('python', 'pythons')
print("The modified sentence is:", modified_sentence)

```

```

⇒ Enter sentence: python
The modified sentence is: pythons

```

```
▶ # accept the score and give the grade based on score
score = float(input("Enter the class score (0-100): "))
if 90 <= score <= 100:
    grade = 'A'
elif 80 <= score < 90:
    grade = 'B'
elif 70 <= score < 80:
    grade = 'C'
elif 60 <= score < 70:
    grade = 'D'
elif 0 <= score < 60:
    grade = 'F'
else:
    grade = 'Invalid score'
print(f"The grade of the class is: {grade}")
```

```
⇒ Enter the class score (0-100): 76
The grade of the class is: C
```

```
[24]
x = [10, 'sachin', 264.98]
types_list = []
for item in x:
    types_list.append(type(item))
print(x)
print(types_list)
```

```
⇒ [10, 'sachin', 264.98]
[<class 'int'>, <class 'str'>, <class 'float'>]
```

```

# Define the sets and list
IT_companies = {'Facebook', 'Google', 'Microsoft', 'Apple', 'IBM', 'Oracle', 'Amazon'}
A = {19, 22, 24, 20, 25, 26}
B = {19, 22, 20, 25, 26, 24, 28, 27}
age = [22, 19, 24, 25, 26, 24, 25, 24]

# 1. Find the length of the set IT_companies
length_IT_companies = len(IT_companies)
print("Length of IT_companies:", length_IT_companies)

# 2. Add 'Twitter' to IT_companies
IT_companies.add('Twitter')
print("IT_companies after adding 'Twitter':", IT_companies)

# 3. Insert multiple IT companies at once to the set IT_companies
additional_companies = {'Salesforce', 'Adobe', 'Nvidia'}
IT_companies.update(additional_companies)
print("IT_companies after adding multiple companies:", IT_companies)

# 4. Remove one of the companies from the set IT_companies
IT_companies.remove('Twitter') # Removing 'Twitter' that was added earlier
print("IT_companies after removing 'Twitter':", IT_companies)

# 5. What is the difference between remove and discard
print("Difference between remove and discard:")
print("-- `remove(element)` raises a `KeyError` if the element is not found in the set.")
print("-- `discard(element)` does not raise an error if the element is not found; it simply does nothing.")

# 6. Join A and B
union_AB = A.union(B)
print("Union of A and B:", union_AB)

# 7. Find A intersection B
intersection_AB = A.intersection(B)
print("Intersection of A and B:", intersection_AB)

# 8. Is A a subset of B?
is_subset = A.issubset(B)
print("Is A a subset of B?", is_subset)

# 9. Are A and B disjoint sets?
are_disjoint = A.isdisjoint(B)
print("Are A and B disjoint sets?", are_disjoint)

# 10. Join A with B and B with A
A.update(B) # Join A with B
print("A after joining with B:", A)

# Reset A and B to their original values
A = {19, 22, 24, 20, 25, 26}
B = {19, 22, 20, 25, 26, 24, 28, 27}

B.update(A) # Join B with A
print("B after joining with A:", B)

# 11. What is the symmetric difference between A and B?
symmetric_difference_AB = A.symmetric_difference(B)
print("Symmetric difference between A and B:", symmetric_difference_AB)

# 12. Delete the sets completely
del IT_companies
del A
del B
age_set = set(age)
print("Length of the list:", len(age))
print("Length of the set:", len(age_set))

```

```
Length of IT_companies: 7
IT_companies after adding 'Twitter': {'Oracle', 'Facebook', 'Google', 'Apple', 'Microsoft', 'IBM', 'Amazon', 'Twitter'}
IT_companies after adding multiple companies: {'Oracle', 'Salesforce', 'Adobe', 'IBM', 'Facebook', 'Amazon', 'Google', 'Apple', 'Nvidia', 'Microsoft', 'Twitter'}
IT_companies after removing 'Twitter': {'Oracle', 'Salesforce', 'Adobe', 'IBM', 'Facebook', 'Amazon', 'Google', 'Apple', 'Nvidia', 'Microsoft'}
Difference between remove and discard:
- `remove(element)` raises a `KeyError` if the element is not found in the set.
- `discard(element)` does not raise an error if the element is not found; it simply does nothing.
Union of A and B: {19, 20, 22, 24, 25, 26, 27, 28}
Intersection of A and B: {19, 20, 22, 24, 25, 26}
Is A a subset of B? True
Are A and B disjoint sets? False
A after joining with B: {19, 20, 22, 24, 25, 26, 27, 28}
B after joining with A: {19, 20, 22, 24, 25, 26, 27, 28}
Symmetric difference between A and B: {27, 28}
Length of the list: 8
Length of the set: 5
```

MY YOUTUBE LINK : <https://youtu.be/DCX6bnQBy18?feature=shared>

MY GITHUB : <https://github.com/PraveenDondapati/bda.git>