

Exercise 10

Programs should be written in the programming language Python according to the respective exercise description. The program must be correct in terms of syntax and semantic. If there exists a minimal solution using only a single pre-defined function, this function is not allowed.

The weekly exercise should be uploaded on ILIAS as a single Jupyter Notebook (.ipynb). Processing time of each exercise runs from Wednesday, 8:00 until following Tuesday, 8:00, if not communicated differently. Make sure that answers to questions are contained within the Jupyter Notebook that is uploaded on ILIAS.

1. Use a list comprehension to create a list named `passing_scores` that includes only the scores that are equal to or greater than 60 from an input list.

Example:

```
scores = [78, 92, 54, 80, 33, 67, 88, 45, 72, 95, 60]
```

Ouptut:

```
<YOUR CODE>
print(passing_scores)
[78, 92, 80, 67, 88, 72, 95, 60]
```

2. You are given a list of strings representing amino acid sequences from proteins. Create a function `process_sequences` that takes the original list of sequences as an argument and returns a new list containing modified versions of the sequences that meet the following criteria: 1. Length filter: Include only sequences with more than 7 amino acids. 2. Transformation: For the selected sequences, convert all amino acids to uppercase. Write the `process_sequences` function using list comprehensions.

Example:

```
original_sequences = [
    "ilmwyiyv",
    "yiywgte",
    "gmpae",
    "tqegpimkn",
    "ptvmcgd",
    "afsp"
]
```

Ouptut:

```
processed_sequences = process_sequences(original_sequences)
print("Processed sequences:", processed_sequences)
Processed sequences: ['ILMWYIYV', 'TQEGPIMKN']
```

3. You are given a list of strings representing amino acid sequences from proteins. Create a dictionary comprehension, that stores for each amino acid as key the sum of occurrences in all of the sequences in the list.

Example:

```
sequences = [
    "ILMWYIYV",
    "YIYWGTE",
    "GMPAE",
    "TQEGPIMKN",
    "PTVMCGD",
    "AFSP"
]
```

Ouput:

```
{'A': 2, 'C': 1, 'D': 1, 'E': 3, 'F': 1, 'G': 4, 'H': 0, 'I': 4, 'K': 1, ... }
```

4. We want to implement a class representing a dice with 6 faces (1, 2, 3, 4, 5, 6). Write a program that defines the respective class `Dice`. Objects of that class should provide the attribute `up_surf` referring to the current upper surface. Provide a `throw` method and a `read` method. The `throw` method should throw the dice by randomly setting the upper surface to a random integer number $1 \leq face \leq 6$. The `read` method should return the value of the current upper surface. Objects of the class should be initialized by throwing the dice.

Example:

```
dice = Dice()
print(dice.read())
dice.throw()
print(dice.read())
dice.throw()
print(dice.read())
```

Ouput:

```
6
5
3
```

5. We want to implement a class representing one playing card out of a standard french 52 cards deck. Objects of the class should define a rank (e.g. "King") and a suit (e.g. "Hearts") variable. The class should also provide a method `show` that returns the values of rank and suit in the form "King of Hearts". The value of the card will be initialized by providing the rank and the suit as arguments in the instantiation call.



Example:

```
card = Card("King", "Hearts")
card.show()
```

Ouptut:

```
'King of Hearts'
```

6. We want to implement a class representing a deck of cards. Instances of the class should initialize the list of 52 cards from the standard french deck. The class should provide a method `shuffle()`, a method `cards()` and a method `draw()`. The method `shuffle()` can be used to shuffle the card deck. The method `cards()` returns the current list of cards. `draw()` draws the top card from the card deck and deletes it from the card deck list.

Example:

```
deck = DeckOfCards()
deck.shuffle()

print("Shuffled deck of cards:")
for card in deck.cards():
    print(card.show())

card = deck.draw()
print("Drawn card from deck:", card.show())

print("Remaining deck of cards:")
for card in deck.cards():
    print(card.show())
```

Ouptut:

```
Shuffled deck of cards:
Ace of Hearts
2 of Clubs
9 of Diamonds
```

¹Modified after <https://upload.wikimedia.org/wikipedia/commons/d/df/4kingsPRL.png>

3 of Diamonds

2 of Diamonds

7 of Hearts

...

Drawn card from deck: Ace of Hearts

Remaining deck of cards:

2 of Clubs

9 of Diamonds

3 of Diamonds

2 of Diamonds

7 of Hearts

...