Roshan Poudel

Part 01

Q4. Looking at the results of the last two commands, we can say that Maps store (key, value) pairs in the non-decreasing order of the keys.

Q5.

my_map = %{“Person A” => 40, “Person B” => 33, “Person C” => 60, “Person D” => 40, “Person E” => 20}

curved grades using mapping

curved_map = Enum.map(my_map, fn {key, value} -> {key, value * 1.3} end)

```
[
  {"Person A", 52.0},
  {"Person B", 42.9},
  {"Person C", 78.0},
  {"Person D", 52.0},
  {"Person E", 26.0}
]
```

Q7. `1..100_000 |> Enum.map(&(&1 * 3)) |> Enum.filter(odd?) |> Enum.sum()`

Generate a list of numbers from 1 to 100000,
Pass that list to the map function that takes the values and multiplies each of them by 3 and return a new list,
Again, pass the list returned by the map function and filter the odd numbers out and return that list,
Now, add all those numbers and return it

7500000000

Part 02

#codepoints

```
String.codepoints("olá")
["o", "l", "á"]

String.codepoints("оптими зации")
["о", "п", "т", "и", "м", "и", " ", "з", "а", "ц", "и", "и"]

String.codepoints("ǎ̓̀Ḥ̣Ω")
["ǎ", "̓̀Ḥ", "Ω̣"]

String.codepoints("\u00e9")
["é"]

String.codepoints("\u0065\u0301")
["e", "́"]


#ends_with?

String.ends_with?("language", "age")
true
String.ends_with?("language", ["youth", "age"])
true
String.ends_with?("language", ["youth", "elixir"])
false


#slice

String.slice("elixir", 1..3)
"lix"
String.slice("elixir", 1..10)
"lixir"

String.slice("elixir", -4..-1)
"ixir"
String.slice("elixir", -4..6)
"ixir"
String.slice("elixir", -100..100)
"elixir"


#split
```

String.split("foo bar")
["foo", "bar"]

String.split("foo" <> <<194, 133>> <> "bar")
["foo", "bar"]

String.split(" foo   bar ")
["foo", "bar"]

String.split("no\u00a0break")
["no\u00a0break"]

```
"$34.56" |> String.split("$") |> List.last |> String.to_float
```

"$34.56" |> String.split("$"): This splits the string at the $ symbol, resulting in a list ["", "34.56"].
|> List.last: Extracts the last element of the list, which is "34.56".
|> String.to_float: Converts the string "34.56" into the float 34.56.