

Exercises making Decisions

Exercise 1

1. create a file named `heads_or_tails.py` in this folder.
2. Write a program that lets the user guess whether the flip of a coin results in heads or tails. The program randomly generates an integer 0 to 1, which represents heads or tails. The program prompts the user to enter a guess and reports whether the guess is correct or incorrect.
 - import the `random` module to generate a random numbers
 - use `random.randint(0, 1)` to generate a random number between 0 and 1
3. The output should look like the following
 - for a correct response

```
$ python heads_or_tails.py
Guess the coin flip! Enter heads or tails (h/t): h
The coin flip was: tails
you guessed wrong!
```

- for an incorrect response

```
$ python heads_or_tails.py
Guess the coin flip! Enter heads or tails (h/t): t
The coin flip was: tails
you guessed correct!
```

1. (optional) test your project by running the following command `python test_do_not_touch/test_heads_or_tails.py`

Exercise 2

1. create a file named `leap_year.py` in this folder.
2. Write a program to determine if a user input year is a leap year. A year is a leap year if it is divisible by 4 but not by 100, or if it is divisible by 400. Here are some sample runs:

```
$ python leap_year.py
Enter a year: 2008
2008 is a leap year? true
```

```
$ python leap_year.py
Enter a year: 1900
1900 is a leap year? false
```

3. (optional) test your project by running the following command `python test_do_not_touch/test_leap_year.py`

Exercise 3

1. create a file named `rock_paper_scissors.py` in this folder.
2. Write a program that plays the scissor-rock-paper game. (A scissor cuts paper, a rock can knock a scissor, and a paper can wrap a rock.) The program randomly generates a number 0, 1, or 2 representing scissor, rock, and paper. The program prompts the user to enter a number 0, 1, or 2 and displays a message indicating whether the user or the computer wins, loses, or draws. Here are sample runs:

- First, you winning

```
$ python rock_paper_scissors.py
Scissor (0), rock (1), paper (2): 1
The computer is scissor. You are rock. You win
```

- Second, you losing

```
$ python rock_paper_scissors.py
Scissor (0), rock (1), paper (2): 2
The computer is scissor. You are paper. You lose
```

- Third a tie

```
$ python rock_paper_scissors.py
Scissor (0), rock (1), paper (2): 0
The computer is scissor. You are scissor. It is a draw
```

Note: again you'll need to use the `random` module `random.randint(0, 2)` to generate a random number between 0 and 2 3.

(optional) test your project by running the following command `python test_do_not_touch/test_rock_paper_scissors.py`

Exercise 4

1. create a file named `zodiac.py` in this folder.
2. Write a program to find out the Chinese Zodiac sign for a given year. The Chinese Zodiac is based on a 12-year cycle, with each year represented by an animal— monkey, rooster, dog, pig, rat, ox, tiger, rabbit, dragon, snake, horse, or sheep —in this cycle. Note `year % 12` determines the Zodiac sign. 1900 is the year of the rat because `1900 % 12` is 4. Here are some sample runs:

```
Enter a year: 1900
rat
```

```
Enter a year: 1887
pig
```

Exercise 5

1. create a file named `month_name.py` in this folder.
2. Write a program that will take a month number from the user and print the name of the month. If the user enters a number that is out of the range 1 to 12, the program should print an error message. Do this using a `match` statement. Here are some sample runs.

```
$ python month_name.py
Enter a month number (1-12): 1
Month is:
January
```

```
$ python month_name.py
Enter a month number (1-12): 4
Month is:
April
```

4. (optional) test your project by running the following command `python test_do_not_touch/test_month_name.py`

Exercise 6

1. Create file named `package_selector.py` in this folder.
2. Write a program for a gym so that it can determine which membership package a person should purchase. There are three packages:
 - Package A: \$40/month, 4 months
 - Package B: \$55/month, 8 months
 - Package C: \$75/month, 12 months
 - Package D: \$100/month, 12 month Note: ensure you have the words "You have selected Package A" or which ever package you select
3. Run you projoust Here are some sample runs:

```
$ python package_selector.py
Welcome to the Ultimate Gym
Please select a membership package:
- Package A: $40/month, 4 months (short-term package)
- Package B: $55/month, 8 months (standard package)
- Package C: $75/month, 12 months (regular package)
- Package D: $100/month, 12 month (premium package, includes 4 free personal training sessions)

Enter the package letter (A/B/C/D): A
You have selected Package A
Your monthly fee is $40
Your total fee is $160
```

Here's a second example

```
$ python package_selector.py
Welcome to the Ultimate Gym
Please select a membership package:
- Package A: $40/month, 4 months (short-term package)
- Package B: $55/month, 8 months (standard package)
- Package C: $75/month, 12 months (regular package)
- Package D: $100/month, 12 month (premium package, includes 4 free personal training sessions)

Enter the package letter (A/B/C/D): D
You have selected Package D
Your monthly fee is $100
Your total fee is $1200
```

4. (optional) test your project by running the following command `python`

```
test_do_not_touch/test_package_selector.py
```