2 Making decisions - solutions to exercises

Table of Contents

2.1 Calculate Body Mass Index (BMI)

Write a Python function named `find_BMI` to calculate body mass index and display the grade from the following grades:

> Underweight: BMI < 18.5
> Normal weight: 18.5 <= BMI < 25.0
> Overweight: 25.0 <= BMI < 30.0
> Obesity (class 1): 30.0 <= BMI < 35
> Obesity (class 2): 35.0 <= BMI

Where weight is taken in kilograms and height in meters. BMI is calculated as your weight W (in kilograms) divided by the square of your height H (in metres) or

$$BMI = \frac{W}{H^2}.$$

In [1]:
```python
def find_BMI(weight, height):
    #calculate the BMI (assuming weight is in kg and height in meters)
    BMI = weight / (height)**2

    if BMI < 18.5:
        print("You are underweight. Your BMI is %.2f" %BMI)
    elif BMI < 25.0:
        print("You are healthy.Your BMI is %.2f" %BMI)
    elif BMI < 30.0:
        print("You are over weight. Your BMI is %.2f" %BMI)
    elif BMI < 35:
        print("You are severely over weight.Your BMI is %.2f" %BMI)
    else:
        print("You are severely obese. Your BMI is %.2f" %BMI)
```

In [2]:
```python
find_BMI(78,1.76)
```

```
You are over weight. Your BMI is 25.18
```

2.2 How many days in the month?

As we know a month may have as few as 28 days, and as many as 31. Write a function named `days` that takes month name, as a parameter, and prints out the number of days in that month. For February, display "28 or 29 days", so that leap year is included.

In [3]:
```
def days(month, leap_year):
    #set it to 31 unless stated otherwise (as per below)
    days = 31

    #find out the number of days
    #note that number of days may be handled as an integer or as a string (whatever is needed/conv

    if month == 'April' or month == 'June' or month == 'September' or month == 'November':
        days = 30
    elif month == 'February':
        if leap_year: days = '29'
        else: days = '29'

    # Display the result
    print(month, 'has', days, 'days in it.')
```

In [4]:
```
days('February', True)
```

```
February has 29 days in it.
```

2.3 Is it the snake year, or the dragon one?

The Chinese zodiac matches animals with years in a 12 year cycle. One such cycle is shown in the table below.

Year	Animal
2000	Dragon
2001	Snake
2002	Horse
2003	Sheep
2004	Monkey
2005	Rooster
2006	Dog
2007	Pig
2008	Rat
2009	Ox
2010	Tiger
2011	Hare

The pattern repeats, and so 2012 was again year of the dragon, then 2013 being the year of the snake etc. Write a function that reads the year from the user (or takes it as a parameter, whatever you prefer) and displays the animal associated with that year. What year would be 2050 or 2099?

In [5]:
```
def zodiac():
    # Read a year from the user
    year = int(input("Enter a year: "))

    # Determine the animal associated with provided year
    if year % 12 == 8: animal = "Dragon"
    elif year % 12 == 9: animal = "Snake"
    elif year % 12 == 10: animal = "Horse"
    elif year % 12 == 11: animal = "Sheep"
    elif year % 12 == 0: animal = "Monkey"
    elif year % 12 == 1: animal = "Rooster"
    elif year % 12 == 2: animal = "Dog"
    elif year % 12 == 3: animal = "Pig"
    elif year % 12 == 4: animal = "Rat"
    elif year % 12 == 5: animal = "Ox"
    elif year % 12 == 6: animal = "Tiger"
    elif year % 12 == 7: animal = "Hare"

    # Report the result
    print("%d is the year of the %s." % (year, animal))
```

In [6]:
```
zodiac()
```

```
Enter a year:  1987

1987 is the year of the Hare.
```

In [7]:
```
zodiac()
```

```
Enter a year:  2008

2008 is the year of the Rat.
```