FINS3646 FINAL EXAM
(z5336994)

Student name: Graciela Evangeline Bahtiar

ZID: z5336994

Instructions:

- Time allowed: 3 hours
- Total marks available: 100 marks
- The final exam is to be completed individually. You must not collaborate with other individuals (students or not).
- Section 1: Multiple choice (4 marks each, 60 marks in total)
  - There are 15 multiple choice questions
  - All multiple-choice questions are graded on a correct/incorrect basis. There is no penalty for answering a multiple-choice question incorrectly.
  - **Ensure that this PDF is for your UNSW ZId**. We will not be able to correct marks should you use the wrong exam paper.
  - Unless otherwise specified, each question is independent of the others and assumptions from one question do not carry over to the others. Unless explicitly stated, assume the questions refer to Python 3.
  - Submit your answers through Ed in the "Multiple choice response" slide. Each question has a Python variable assigned to it. Assign a string with your answer to the question variable. For example, to answer "b" for Question 4, use the code `q04 = "b"`. Ed has been pre-populated with variable names to save you time.
  - **Click on Mark in Ed** to save your progress and submit your answers. Your responses will not be recorded if you fail to click on *Mark*.
- Section 2: Coding challenges (total of 30 marks)
  - The code challenge has three functions you need to complete (each worth 1/3 of the marks for the challenge)
  - You must submit all code challenges through ED.
- Section 3: Short answers (total of 10 marks)
  - There are two questions in this section (each worth 5 marks).
  - Please make sure you submit your answers before the end of the exam.

Question 1:

If `df` is a non-empty data frame, then `df.iloc[:, 0]` has the same index as `df`.

a. True
b. False

Question 2:

Suppose the data frames `df1` and `df2` have no index label in common and no column label in common. Neither `df1` nor `df2` are empty data frames. Given the following statements:

```
df3 = df1.join(df2, how='left')
df4 = df2.join(df1, how='right')
```

Which of the following is true?

a. All elements in `df3` are NaN
b. The data frames `df3` and `df4` are both empty
c. All elements in `df4` are NaN
d. The `df3` data frame is empty
e. None of the above

Question 3:

Consider the following piece of code:

```
i = 0
while i < 5:
    print(i)
    i += 1
    if i == 3:
        break
else:
    print(0)
```

This program will output the following numbers (in this order):

a. 1 2 3
b. 0 1 2 3
c. 1 2 0
d. 1 2
e. 0 1 2

Question 4:

Let `df` be any pandas data frame. The statement `df.loc[:, ['col1', 'col2']]` will **always** return a data frame.

a. True
b. False

Question 5:

What is the object assigned to `x` after the following statements are executed?

```
y = int(98.6) + 1.0
x = f'{int(y)}'
```

a. A `str` with value "100"
b. An `int` with value 99
c. A `str` with value "99"
d. An `int` with value 100
e. None of the above

Question 6:

Consider the following function:

```
def func(par):
    lst = []
    def sub_func(lst):
        lst.append(hidden_function(par))
        return lst
    return lst

res = func(1)
```

where `hidden function` is a properly defined and valid Python function, which accepts an integer as a parameter, does not include references to the name `lst`, and whose name is available in the global namespace. Assume that the code above will not raise any Exceptions.

In the code above, what is the object assigned to `res`?

a. Not possible to determine
b. A list containing the output of `hidden_function(1)`
c. `[]`
d. `[None]`
e. None of the above

Question 7:

During the execution of the code below, how many instances are created?

```
x = [1, 2, 3]
y = x
```

a. 6
b. 1
c. 4
d. 3
e. 2

Question 8:

Suppose your `toolkit` project folder (in PyCharm) includes the following files:

```
toolkit/
|
|__ main.py
|
|__ lines.txt
```

Module `main.py` includes the following statement(s):

```
with open('lines.txt') as fobj:
    lines = fobj.read()

# >>
```

Suppose the code above does not raise any exceptions when executed. At the point indicated in the code above by the `# >>` line, the `fobj` variable does not exist in the global namespace.

a. True
b. False

Question 9:

Suppose your `toolkit` project folder (in PyCharm) includes the following files:

```
toolkit/
|
|__ mod0.py
|
|__ pkg1/
|   |
|   |__ __init__.py
|   |
|   |__ mod1.py
|   |
|   |__ pkg2/
|   |   |
|   |   |__ __init__.py
|   |   |
|   |   |__ mod2.py
```

Module `mod0.py` includes the following statement(s):

```
import pkg1.mod1
print(pkg1.pkg2.mod2.var)
```

Module `mod1.py` includes the following statement(s):

```
from pkg1 import pkg2
```

Module `mod2.py` includes the following statement(s):

```
var = 'here'
```

Which of the following will **not** happen when we execute the `mod0.py` module?

a. The `from pkg1 import pkg2` statement will execute without an exception
b. The `import pkg1.mod1` statement will execute without an exception
c. The name `mod2` will not be added to the global namespace
d. The statements in `mod2.py` will not be executed
e. The string `here` will be printed

Question 10:

Which of the following are not valid variable names in Python?

- (1): `my_string_1`
- (2): `1st_string`
- (3): `foo`
- (4): `_`
- (5): `__init__`

a. Only (1), (3), (4), (5)
b. Only (2), (5)
c. Only (2)
d. Only (2), (4), (5)
e. Only (1), (3)

Question 11:

Which of the following options is equivalent to `x = [5, 5, 5, 5, 5]`?

Option 1:

```
x = [5, 5]
x.extend([5, 5, 5])
```

Option 2:

```
x = [5] * 5
```

Option 3:

```
x = [5]
for i in range(5):
    x += [5]
```

Option 4:

```
x = [5, 5, 5, 5, 5, 5]
x.remove(5)
```

a. Option 2
b. Options 1, 2 & 3
c. Options 1 & 3
d. Options 1, 2 & 4
e. Options 1 & 2

Question 12:

Suppose `df` is a data frame with 1,000 rows. The output of `print(df)` is:

```
       ticker  return
index
1999        A     0.1
2000        A     0.1
...       ...     ...
2001        B     0.4
2000        B     0.4
```

where `index` is the name of the index, 'ticker' and 'return' are column labels, and '...' represents hidden rows.

Which of the following expressions will return a data frame with the most recent return for each ticker?

a. `df.loc[:, ['ret']].groupby('ticker').last()`
b. `df.loc[:, ['ticker']].groupby('ticker').last()`
c. `df.loc[:, ['ret']].apply(last())`
d. `df.groupby('ticker').last()`
e. None of the above

Question 13:

Suppose `hidden function` is a properly defined and valid Python function, which accepts a string as a parameter, returns a string as a result, and whose name is available in the global namespace. Assume that the function call `hidden_function('a proper string')` will not raise an exception.

In the code below, what will be the object assigned to the variable `res`?

```
def func(var2):
    if var2 is None:
        var1 = hidden_function(var1)
    else:
        var1 = 'alternative'
    return var1


var1 = 'A'
var2 = None
res = func(var2)
```

a. A `str` with value "alternative"
b. Python will raise an Exception
c. A `str` with value "A"
d. A `str` with the output of `hidden_function("A")`
e. None of the above

Question 14:

If `df` is a non-empty data frame, then `df.iloc[:, 0]` will return a series.

a. False
b. True

Question 15:

Consider the following code:

```
import datetime as dt
import pandas as pd

dates = [
    '2020-01-01',
    '2020-01-02',
    '2020-01-03',
]
col0 = [
    1,
    10,
    2,
]


df0 = pd.DataFrame(col0, index=dates, columns=['col0'])
df0.index.name = 'date'

df1 = pd.to_datetime(df0.loc[:, 'col0'])

df2 = df0.reset_index()
df2.index = pd.to_datetime(df0.index)

df3 = pd.DataFrame(col0, index=pd.to_datetime(dates))
```

Which of the following statements is true?

a. `df0.index` is **not** a `DatetimeIndex`
b. `df1` is not a data frame
c. `df3.index` is a `DatetimeIndex`
d. `df2.index` is a `DatetimeIndex`
e. All the above statements are true