

International Information Technology University JSC
Faculty of Information Technology
Department of Mathematical and Computer Modeling



SYLLABUS (ACADEMIC PROGRAM)

Course (code, title): PRAD 5207 R/Python for data analysis

Major (code, title): 7M061 Information and Communication Technologies

Educational program 7M06106 Data Science

Year: 1; **Semester:** 1; **Number of credits:** 5 ECTS

Lectures: 15 hours

Laboratory classes: 30 hours

T/SIS: 105 hours

Total: 150 hours

Final assessment form: Examination

Almaty 2020

Academic program of the course PRAD 5207 R/Python for data analysis has been reviewed at the meeting of Mathematical and Computer Modelling department.

Minutes № 1 dated «17» August 2020

Head of the Department



Ydyrys A.Zh., Assistant professor, PhD

Author



Nurtas M., assistant professor, PhD

The working academic program was approved at the meeting of the Educational and Methodological Board of JSC "IITU"

Minutes № 1 dated "28" August 2020

Director of the Department
of Postgraduate Education



signature

G.U. Bektemisova

1. GENERAL INFORMATION	
Faculty	Information Technology
Major code and title	7M061 Information and Communication Technologies
Educational program code and title	7M06106 Data Science
Year, semester	1 st year, 1 st semester
Subject category	Basic
Number of credits (ECTS)	5
Prerequisites	Algorithms and Programming Languages, Linear Algebra, Discrete Mathematics, Introduction to Probability and Statistics
Postrequisites	Machine learning 1
Lecturer	Marat Nurtas., PhD, assistant professor., 807 room, m.nurtas@iitu.kz office hours: Monday 12:10-13:00
2. GOALS, OBJECTIVES AND LEARNING OUTCOMES OF THE COURSE	
The course goal is	
<p>Students with a minor in Introduction to Programming in Python. Students will develop relevant programming abilities. Students will demonstrate proficiency with visualization of data. Students will develop the ability to build and assess data-based models. Students will execute statistical analyses with professional statistical software. Students will demonstrate skill in data management.</p>	
The objectives of the course are	
<ul style="list-style-type: none"> - The orientation for decision-making. - The credibility of the mathematical model in Financial Mathematics and Data science. - The necessity of using a computer. 	
Learning outcomes of the course	
<p>Students successfully completing the course will be able to:</p> <ul style="list-style-type: none"> - Learn about powerful ways to store and manipulate data, and helpful data science tools to begin conducting analyses. - Learn to visualize real data with Matplotlib's functions and get acquainted with data structures such as the dictionary and the pandas DataFrame. - After covering key concepts such as boolean logic, control flow, and loops in Python, students will be ready to blend together everything students has learned to solve a case study using hacker statistics. - You'll come out of this course being able to write your very own custom functions, complete with multiple parameters and multiple return values, along with default arguments and variable-length arguments. 	
3. Course description	
<p>Learn how to analyze data using Python. This course will take you from the basics of Python to exploring many different types of data. You will learn how to prepare data for analysis, perform simple statistical analysis, create meaningful data visualizations, predict future trends from data, and more! Data Analysis with Python will be delivered through lecture, lab, and assignments. It includes following parts: Data Analysis libraries: will learn to use Pandas, Numpy and Scipy libraries to work with a sample dataset. We will introduce you to pandas, an open-source library, and we will use it to load, manipulate, analyze, and visualize cool datasets. Then we will introduce you to another open-source library, scikit-learn, and we will use some of its machine learning algorithms to build smart models and make cool predictions.</p>	
4. COURSE POLICY	
<p>Students are forbidden to:</p> <ul style="list-style-type: none"> - submit any tasks after the deadline. The mark for late submissions is decreased; - cheat. Plagiarized papers shall not be graded; 	

- be late for classes. Three times' tardy amounts to one absence;
- retake any tests, unless there is a valid reason for missing them;
- use mobile phones in class;
- chew gum in class.

Students should always

- be appropriately dressed (formal/semi- formal styles are acceptable);
- show consideration for and mutual support of teachers and other students;
- let the teacher know of any problems arising in connection with English studies.

Online classes will be on platforms MS Teams, DL and Zoom.

5. LITERATURE

Basic literature:

1. Joel G. Python for Data Science, O'Reilly Media 2016.
2. Joel G. Data Science from Scratch, O'Reilly Media 2015.
3. https://www.coursera.org/programs/international-information-technologies-university-on-coursera-x7m8z/browse?currentTab=CATALOG&productId=ATH_wZoBEeqNYhKXUrAO4w&productType=course&query=introduction+to+python&showMiniModal=true
4. https://www.coursera.org/programs/international-information-technologies-university-on-coursera-x7m8z/browse?currentTab=CATALOG&productId=F-h1g0w7EeWeOApO_15R1w&productType=s12n&query=python&showMiniModal=true

Supplementary literature:

1. Introduction to Programming in Python, Durham University 2014.

6. Course schedule

Week No	Course Topic	Reference Materials	Lectures (1 h/w)	Laboratory sessions (2 h/w)	TSIS (1 h/w)	SIS (6 h/w)
1	Module 1. Introduction. Variables, Types, Operators, Checks, Loops.	[1]-[2]	1	2	1	6
2	Module 2. Work with files and complex metrics	[1]-[2]	1	2	1	6
3	Module 3. Introduction to Pandas	[1]-[2]	1	2	1	6
4	Module 4. Functions and Groupings	[1]-[3]	1	2	1	6
5	Module 5. PivotTables and Analytics	[2]-[3]	1	2	1	6
6	Module 6. Merging Data Frames	[1]-[2]	1	2	1	6
7	Module 7. Work with multiple files Mid term	[1]-[3]	1	2	1	6
8	Module 8. Work with dates	[1]-[2]	1	2	1	6
9	Module 9. Visualizations and reports	[1]-[2]	1	2	1	6

10	Module 10. Merging large dataframes	[1]-[2]	1	2	1	6
11	Module 11. Work with HTML pages and API VKontakte	Use website	1	2	1	6
12	Module 12. Yandex Services API	Use website	1	2	1	6
13	Module 13. Google Services API	Use website	1	2	1	6
14	Module 14. Online reporting with google sheet	Use website	1	2	1	6
15	Module 15. Work with bugs and bots	Use website	1	2	1	6
Total hours		150 hours	15	30	15	90

7. List of topics/ assignments for laboratory classes

№	Topic Title	Number of hours	References	Form of reporting	Deadline
1	2	3	4	5	6
1.	Module 1. Introduction. Variables, Types, Operators, Checks, Loops.	2	[2], pp. 13-15	Submission to DL	8.09.20
2.	Module 2. Work with files and complex metrics	2	[2], pp. 16-18 http://www.python.org/doc/	Submission to DL	15.09.20
3.	Module 3. Introduction to Pandas	2	[2], pp. 25-26	Submission to DL	22.09.20
4.	Module 4. Functions and Groupings	2	[2], pp. 35-38	Submission to DL	29.09.20
5.	Module 5. PivotTables and Analytics	2	[2], pp. 40- 45	Submission to DL	06.10.20
6.	Module 6. Merging Data Frames	2	[2], pp. 55-56	Submission to DL	13.10.20
7.	Module 7. Work with multiple files Mid term	2	[2], p. 60	Submission to DL	20.10.20
8.	Module 8. Work with dates	2	[2], p. 67	Submission to DL	27.10.20
9.	Module 9. Visualizations and reports	2	[2], pp. 67-69	Submission to DL	03.11.20

10.	Module 10. Merging large dataframes	2	[2], pp. 73-79	Submission to DL	10.11.20
11.	Module 11. Work with HTML pages and API VKontakte	2	[2], pp. 13-15	Submission to DL	17.11.20
12.	Module 12. Yandex Services API	2	[2], pp. 16-18 http://www.python.org/doc/	Submission to DL	24.11.20
13.	Module 13. Google Services API	2	[2], pp. 25-26	Submission to DL	04.12.20
14.	Module 14. Online reporting with google sheet	2	[2], pp. 35-38	Submission to DL	04.12.20
15.	Module 15. Work with bugs and bots	2	[2], pp. 40- 45	Submission to DL	11.12.20

8. List of topics/ assignments for practical classes

№	Topic Title	Number of hours	References	Form of reporting	Deadline
1	2	3	4	5	6
1.	Module 1. Introduction. Variables, Types, Operators, Checks, Loops.	1	[2], pp. 13-15	Submission to DL	8.09.20
2.	Module 2. Work with files and complex metrics	1	[2], pp. 16-18 http://www.python.org/doc/	Submission to DL	15.09.20
3.	Module 3. Introduction to Pandas	1	[2], pp. 25-26	Submission to DL	22.09.20
4.	Module 4. Functions and Groupings	1	[2], pp. 35-38	Submission to DL	29.09.20
5.	Module 5. PivotTables and Analytics	1	[2], pp. 40- 45	Submission to DL	06.10.20
6.	Module 6. Merging Data Frames	1	[2], pp. 55-56	Submission to DL	13.10.20
7.	Module 7. Work with multiple files Mid term	1	[2], p. 60	Submission to DL	20.10.20

8.	Module 8. Work with dates	1	[2], p. 67	Submission to DL	27.10.20
9.	Module 9. Visualizations and reports	1	[2], pp. 67-69	Submission to DL	03.11.20
10.	Module 10. Merging large dataframes	1	[2], pp. 73-79	Submission to DL	10.11.20
11.	Module 11. Work with HTML pages and API VKontakte	1	[2], pp. 13-15	Submission to DL	17.11.20
12.	Module 12. Yandex Services API	1	[2], pp. 16-18 http://www.python.org/doc/	Submission to DL	24.11.20
13.	Module 13. Google Services API	1	[2], pp. 25-26	Submission to DL	04.12.20
14.	Module 14. Online reporting with google sheet	1	[2], pp. 35-38	Submission to DL	04.12.20
15.	Module 15. Work with bugs and bots	1	[2], pp. 40- 45	Submission to DL	11.12.20

9. List of assignments for Student Independent Study

Proper organization of students independent study is the key to the formation of skills in mastering, learning, assimilation and systematization of acquired knowledge, ensuring a high level of academic performance in the learning process

№	Assignments (topics) for Independent study	Hours	Recommended literature and other sources (links)	Form of submission
1	2	3	4	5
1	Assignments, Strings and Types	2	[2], pp. 13-15	report
2	Errors and Exceptions	2	[2], pp. 16-18 http://www.python.org/doc/	report
3	Exercises 5.1-5.4	2	[2], pp. 25-26	report
4	Indexing Sequences - Summary Exercises 6.1-6.5	2	[2], pp. 35-38	report
5	Exercises 7.1-7.5	2	[2], pp. 40- 45	report
6	Exercises 8.1-8.4	2	[2], pp. 55-56	report
7	Exercise 9.1 – <i>Advanced Topic</i>	2	[2], p. 60	report
8	The SciPy Module – <i>Advanced Topic</i>	2	[2], p. 67	report
9	Exercises 10.1-10.2	2	[2], pp. 67-69	report
10	Exercises 11.1-11.4	2	[2], pp. 73-79	report

10. System for evaluating student performance in a discipline:

Period	Assignments	Number of points	Total
1 st attestation	Student Independent Study / laboratories: 1 Lab, 2 Lab, 3 Lab, 4 Lab, 5 Lab, 6 Lab, Quiz: 1 quiz Mid term	100 each 30 % 20% 50%	100
2 nd attestation	Student Independent Study / laboratories: 1 Lab, 2 Lab, 3 Lab, 4 Lab, 5 Lab, 6 Lab, Quiz: 2 quiz End of term	100 each 30 % 20% 50%	100
Final exam	Exam	100	100
Total	0,3*1stAtt+0,3*2ndAtt+0,4*Final		100

*If the number of absences exceeds 20%, student will be automatically scheduled for a Retake (summer semester)

11. Assessment criteria:

The point-rating letter system for assessing the educational achievements of students with their interpretation in the traditional grading scale:

Letter Grade	Numerical equivalent	Points (%)	Traditional system assessment	General description of grading criteria
A	4,0	95-100	Excellent	The student has knowledge of the subject in the full scope of the curriculum, understands the discipline deeply enough; shows a high level of knowledge that exceeds the volume provided by the syllabus, gives an exhaustive answer
A-	3,67	90-94		The student has knowledge of the subject in the full scope of the curriculum, understands the discipline deeply enough; gives an exhaustive answer
B+	3,33	85-89	Good	The student shows a complete, well-founded knowledge of the subject, but the
B	3,0	80-84		

B-	2,67	75-79		answers did not always highlight the main idea, rational methods of calculation were not always used; the answers were mostly brief and sometimes unclear.
C+	2,33	70-74		
C	2,0	65-69	Satisfactory	The student demonstrates sufficient knowledge of the subject, but without proper depth and justification, the answers are unclear and without proper logical sequence.
C-	1,67	60-64		
D+	1,33	55-59		
D	1,0	50-54		
FX	0,5	25-49	Unsatisfactory	The student demonstrates insufficient knowledge of the subject, positive answers were not given to individual questions.
F	0	0-24		The student demonstrates a very low level of knowledge of the subject.

12. Assessment and evaluation materials (exam questions):

Assignment for the Midterm:

- a) Write a Python program to construct the following pattern, using a nested for loop.

```
*
**
***
****
*****
****
***
**
*
```

- b) Write a Python program to count the number of even and odd numbers from a series of numbers.

Sample numbers : numbers = (1, 2, 3, 4, 5, 6, 7, 8, 9)

Expected Output :

Number of even numbers : 5

Number of odd numbers : 4

- c) Write a Python program which takes two digits m (row) and n (column) as input and generates a two-dimensional array. The element value in the i-th row and j-th column of the array should be $i*j$.

Note :

$i = 0, 1, \dots, m-1$

$j = 0, 1, \dots, n-1$.

Test Data : Rows = 3, Columns = 4

Expected Result : [[0, 0, 0, 0], [0, 1, 2, 3], [0, 2, 4, 6]]

Assignment for the End term:

- a) Write a Python program to sort a list of elements using the merge sort algorithm.

Note: According to Wikipedia "Merge sort (also commonly spelled mergesort) is an $O(n \log n)$ comparison-based sorting algorithm. Most implementations produce a stable sort, which

means that the implementation preserves the input order of equal elements in the sorted output."

- b) Write a Python program to sort a list of elements using the quick sort algorithm.
Note: According to Wikipedia "Quicksort is a comparison sort, meaning that it can sort items of any type for which a "less-than" relation (formally, a total order) is defined. Inefficient implementations it is not a stable sort, meaning that the relative order of equal sort items is not preserved. Quicksort can operate in-place on an array, requiring small additional amounts of memory to perform the sorting."

- c) Write a Python program to create an instance of an OrderedDict using a given dictionary. Sort the dictionary during the creation and print the members of the dictionary in reverse order.

Expected Output:

Angola 244

Andorra 376

Algeria 213

Afghanistan 93

Albania 355

In reverse order:

Albania 355

Afghanistan 93

Algeria 213

Andorra 376

Angola 244

Assignment for the End term:

- a) Write a Python program to calculate the sum of a list of numbers.

- b) Write a Python program for binary search for an ordered list.

Test Data :

Ordered_binary_Search([0, 1, 3, 8, 14, 18, 19, 34, 52], 3) -> True

Ordered_binary_Search([0, 1, 3, 8, 14, 18, 19, 34, 52], 17) -> False

- c) Write a Python program to iterate over an enum class and display individual member and their value.

Expected Output:

Afghanistan = 93

Albania = 355

Algeria = 213

Andorra = 376

Angola = 244

Antarctica = 672

Note: each student will have own tasks. Examples above just for single case.