

# **INTRODUCTION TO PROGRAMMING**

### Grado en Administración de Empresas / Bachelor in Business Administration BBA SEP-2024 PRO-NBA.1.S.E

Area Information Systems and Technology Number of sessions: 20 Academic year: 24-25 Degree course: FIRST Number of credits: 3.0 Semester: 2° Category: COMPULSORY Language: English

# Professor: MARIANO ALVAREZ DIENTE

E-mail: mad2@faculty.ie.edu

### Academic Background

- Graduate and Master in Telecommunication Engineer, Universidad de Valladolid
- Graduate and Master in Literary Theory and Comparative Literature, Universidad de Valladolid.
- Master in E.S.O, Bachillerato, Professional training and languages teaching.
- Master in Philological studies: professional/business applications.

### Academic experience

- Bachelor in Business Administration at IE University. Since 2010.
- Bachelor in Information Systems and Computer Science. Since 2017.
- Currently, Instructor at IBM Global Services on Internetworking and Information and Telecommunication Technologies. Since 2004.

### Courses taught at IBM:

- Big Data & Analytics
- Mobile & corporate strategy
- New technologies in communications
- WLAN networks implementation
- WLAN security
- WiMAX technology
- TCP/IP fundamentals
- Networks Implementation & administration
- IP Telephony (VoIP).
- IoT (IBM Cloud)

### **Professional experience**

- Currently. Freelance. Telecommunication and Information Technologies projects.
- Currently. Instructor at IBM Global Services

- Engineer in Acoustics. Acoustic municipal centre Hall of Valladolid
- Solyven S.L. Solar energy projects.

#### Office Hours

Office hours will be on request. Please contact at:

Contact details: e-mail: mad2@faculty.ie.edu

Office hours: Live tutorials available by previous appointment.

### SUBJECT DESCRIPTION

Have you ever wondered how a computer works? Or how to develop the software your new start-up company needs?

Computers are one of the most configurable machines we humans have invented since the dawn of time yet most of its users are constrained to uses designed by others (third party software). In this course you will learn how to code computer programs that will allow you to expand your computer's functionality up to its full potential.

Knowing how to code is not something reserved for engineers or experts, everyone can write their own programs and if you combine your acquired business knowledge with the ability to write great software programs you could unleash a new set of opportunities for your career and boost your performance as an entrepreneur. If you are interested in knowing how to code software programs this course is for you.

This is a course for beginners! Students joining this course do not need to have any previous coding experience as the course is designed for students learning how to code for the first time.

### LEARNING OBJECTIVES

The main objectives that the student will get are the following:

- Develop a logical thinking by carrying out programs.
- Be able to strengthen problem-solving skills which has been identified as one of the critical skills that will be needed in the job market in the years to come.
- Get strong capabilities in programming with Python.

In this course you will learn programming terminology and will obtain a solid grasp of the basic mechanics of programming. This includes:

- Introduction to problem solving for programming (i.e., "how to think about solving the problem" including techniques such as pseudo-code or flowcharts).

- Understand object-oriented programming and its importance in writing business software applications.

- You will be able to write fully functional console or GUI python programs.
- Basics of data science tasks in Python.

There are also solid reasons why it is important for business students and future managers to learn the basics of programming.

1. Understanding programming concepts can help managers communicate more effectively with their technical teams. This can be especially useful when discussing project requirements and technical issues.

2. Programming skills can help managers become more analytical in their decision making.

Knowing the basics of Python helps better understand the built-in tools for data analysis and visualization, which can be used to inform business decisions. These aspects will be deepened in other courses of the program and this present course introduces the notions and concepts to adequately address them.

3. Programming can be a useful problem-solving skill. Many business problems can be

expressed in a way that is amenable to solution through programming, and being able to write code can help managers to identify and implement solutions more efficiently.

4. Learning programming can be a valuable way for managers to broaden their knowledge and remain competitive in an increasingly technical business environment.

5. As a manager, it is your responsibility to improve processes to add value to the customer or make the company more profitable. This will require new and improved software that automates bureaucracy and provides an improved way for the customer to interact with the company. The manager must be able to understand the potential of coding to make these improvements. The manager must also be able to communicate these needs to the developers so that they can translate them into code that meets the requirements.

6. Programming requires that you first structure what you want to accomplish, determine how to get there and be able to express it in a very clear way. In other words, programming helps you think in an analytical way.

# TEACHING METHODOLOGY

IE University teaching method is defined by its collaborative, active, and applied nature. Students actively participate in the whole process to build their knowledge and sharpen their skills. Professor's main role is to lead and guide students to achieve the learning objectives of the course. This is done by engaging in a diverse range of teaching techniques and different types of learning activities such as the following:

Learning Activity	Weighting	Estimated time a student should dedicate to prepare for and participate in	
Lectures	21.3 %	16.0 hours	
Discussions	10.7 %	8.0 hours	
Exercises in class, Asynchronous sessions, Field Work	37.3 %	28.0 hours	
Group work	10.7 %	8.0 hours	
Individual studying	20.0 %	15.0 hours	
TOTAL	100.0 %	75.0 hours	

### **AI POLICY**

#### #3 - Restricted use of GenAl

In today's world, generative artificial intelligence (GenAI) is changing how we work, study and, in general, how we get things done. However, in the context of this course, the use of GenAI is not permitted, unless it is otherwise stated by the instructor. The use of GenAI tools would jeopardize the students' ability to acquire fundamental knowledge or skills of this course.

If a student is found to have used AI-generated content for any form of assessment, it will be considered academic misconduct, and the student might fail the respective assignment or the course.

### PROGRAM

### **SESSION 1 (LIVE IN-PERSON)**

### Course introduction and overview of the computer programming environment

In this session we will review the course logistics and organization together with the computer programming environment (IDEs, compilers, etc...)

Live classes / lectures

Slides and materials will be provided by the professor

### **SESSION 2 (LIVE IN-PERSON)**

#### Data types & variables

In this session we will learn how to store information in our programs using variables and what are the different data-types for variables in Python. We will program our first "Hello World" program in the Python language.

Live classes / lectures

Slides and materials will be provided by the professor

### **SESSION 3 (LIVE IN-PERSON)**

#### Writing Programs

In this session, we will learn the process for planning and coding software.

Live classes / lectures

Slides and materials will be provided by the professor

### **SESSION 4 (LIVE IN-PERSON)**

#### Definite Loops and the Math and Random libraries

In this session we will introduce our first loops and the use of the Math and Random libraries.

Live classes / lectures

Slides and materials will be provided by the professor

### **SESSION 5 (LIVE IN-PERSON)**

### **Practical session**

In this session we will practice with more exercises based on the content covered in the previous sessions (1 to 4).

Live classes / lectures

Slides and materials will be provided by the professor

# **SESSION 6 (LIVE IN-PERSON)**

### **Decision Structures Part 1**

In this session we will expand the functionality we can code with iterations, logic and flow control with if/else structures.

Live classes / lectures

Slides and materials will be provided by the professor

# **SESSION 7 (LIVE IN-PERSON)**

### **Decision Structures Part 2**

In this session we will expand the functionality we can code with iterations, logic and flow control with if/else structures. Exception Handling.

Live classes / lectures

Slides and materials will be provided by the professor

# **SESSION 8 (LIVE IN-PERSON)**

### **Indefinite Loops**

In this session we will expand the functionality we can code with iterations, logic and flow control with indefinite loops.

Live classes / lectures

Slides and materials will be provided by the professor

# **SESSION 9 (LIVE IN-PERSON)**

### **Practical session**

In this session we will practice with more exercises based on the content covered in the previous sessions (6 to 8).

### Live classes / lectures

Slides and materials will be provided by the professor

### **SESSION 10 (LIVE IN-PERSON)**

### **Boolean Algebra**

In this session we will expand the functionality we can code with the boolean datatype and Boolean algebra.

Live classes / lectures

Slides and materials will be provided by the professor

# **SESSION 11 (LIVE IN-PERSON)**

### Functions, arguments and return types

In this session we will start encapsulating our code in methods that enable us to reuse the code preventing duplication and errors in large projects.

Live classes / lectures

Slides and materials will be provided by the professor

# **SESSION 12 (LIVE IN-PERSON)**

### **Lists and Strings**

In this session we will learn how to store multiple items in lists in Python and process text (strings).

Live classes / lectures

Slides and materials will be provided by the professor

# **SESSION 13 (LIVE IN-PERSON)**

### **Practical session**

In this session we will practice with more exercises based on the content covered in the previous sessions (10 to 12).

Live classes / lectures

Slides and materials will be provided by the professor

# **SESSION 14 (LIVE IN-PERSON)**

### **Reading and Writing Files**

In this session we will learn how to use files to store information on the long term memory.

Live classes / lectures

Slides and materials will be provided by the professor

# **SESSION 15 (LIVE IN-PERSON)**

### **GUI programming**

In this session we will use the graphic library form PYTHON to develop our first simple GUI based program.

Live classes / lectures

Slides and materials will be provided by the professor

# **SESSION 16 (LIVE IN-PERSON)**

### **GUI programming**

In this session we will use the graphic library form PYTHON to develop our first simple GUI based program.

Live classes / lectures

Slides and materials will be provided by the professor

# **SESSION 17 (LIVE IN-PERSON)**

### **Practical session**

In this session we will practice with more exercises based on the content covered in the previous sessions (14 to 16).

Live classes / lectures

Slides and materials will be provided by the professor

# **SESSION 18 (LIVE IN-PERSON)**

### Practical session

In this session we will practice exercises with new tools

Live classes / lectures

Slides and materials will be provided by the professor

# **SESSION 19 (LIVE IN-PERSON)**

#### **Practical Session**

This session will be dedicated to coding some practical exercises, which will involve making use of some of the most important topics of the course.

Live classes / lectures

Slides and materials will be provided by the professor

# **SESSION 20 (LIVE IN-PERSON)**

Final Exam

# **EVALUATION CRITERIA**

### 1. Class participation – discussion

You are expected to attend every class and participate in the discussions and class activities (games, etc.). The basic criteria in grading your participation are: a) your active presence in each session, b) your (quality) contributions to the group discussion c) submission of activities. Lively discussions in the classroom are always encouraged, however, make sure that you provide constructive comments which contribute to the learning experience of the whole class.

#### 2. Group assignment

A major part of this course's learning experience consists of a group project which will be focusing on designing and implementing an algorithmic solution to a business problem.

The definition of the problem as well as the main requirements will be provided to you during the sessions, so that you have only to focus on designing and implementing the best solution. Remember, creativity is always rewarded!

#### 3. Final exam

At the end of the course you will have to pass an individual exam. A minimum of 3.5 out of 10 on the final exam will be required to pass the course. This means that those students who obtain a grade lower than 3.5 in the final exam and their total final grade is higher than 5 out of 10, will still fail the course.

criteria	percentage	Learning Objectives	Comments
Final Exam	40 %		
Intermediate tests	20 %		
Group Work	30 %		
Class Participation	10 %		

### **RE-SIT / RE-TAKE POLICY**

Each student has four (4) chances to pass any given course distributed over two (2) consecutive academic years. Each academic year consists of two calls: one (1) ordinary call (during the semester when the course is taking place); and one (1) extraordinary call (or "re-sit") in June/July.

Students who do not comply with the attendance requirement in each subject during the semester will automatically fail both calls (ordinary and extraordinary) for that Academic Year and have to retake the course (i.e., re-enroll) during the next Academic Year.

The Extraordinary Call Evaluation criteria will be subject to the following rules:

- Students failing the course in the ordinary call (during the semester) will have to re-sit

evaluation for the course in June / July (except those students who do not comply with the attendance rule, and therefore will not have that opportunity, since they will fail both calls and must directly re-enroll in the course during the next Academic Year)

- It is not permitted to change the format nor the date of the extraordinary call exams or deadlines under any circumstance. All extraordinary call evaluation dates will be announced in advance and must be taken into consideration before planning the summer (e.g. internships, trips, holidays, etc.)
- The June/July re-sit will consist of a comprehensive evaluation of the course. Your final grade for the course will depend on the performance in this exam or evaluation only. I.e., continuous evaluation over the semester (e.g. participation, quizzes, projects and/or other grade components over the semester) will not be taken into consideration on the extraordinary call. Students will have to achieve the minimum passing grade of 5 and the maximum grade will be capped at 8.0 (out of 10.0) i.e., "notable" in the extraordinary call.
- Re-takers: Students who failed the subject on a previous Academic Year and are now reenrolled as re-takers in a course will need to check the syllabus of the assigned professor, as well as contact the professor individually, regarding the specific evaluation criteria for them as re-takers in the course during that semester (ordinary call of that Academic Year).

The maximum grade that may be obtained as a retaker during the ordinary call (i.e., the 3rd call) is 10.0 (out of 10.0).

After exams and other assessments are graded by the professor (on either the ordinary or extraordinary call), students will have a possibility to attend a review session (whether it be a final exam, a final project, or the final overall grade in a given course). Please be available to attend the session in order to clarify any concerns you might have regarding your grade. Your professor will inform you about the time and place of the review session.

Students failing more than 18 ECTS credits after the June/July re-sits will be asked to leave the Program. Please, make sure to prepare yourself well for the exams in order to pass your failed subjects.

In case you decide to skip the opportunity to re-sit for an exam or evaluation during the June/July extraordinary call, you will need to enroll in that course again for the next Academic Year as a re-taker, and pay the corresponding tuition fees. As you know, students have a total of four (4) allowed calls to pass a given subject or course, in order to remain in the program.

# **BIBLIOGRAPHY**

### Recommended

- Zelle, John. (2017). *Python Programming: An introduction to computer Science.*. 3rd. Franklin, Beedle. ISBN 9781590282755 (Printed)

### **BEHAVIOR RULES**

Please, check the University's Code of Conduct <u>here</u>. The Program Director may provide further indications.

### ATTENDANCE POLICY

Please, check the University's Attendance Policy <u>here</u>. The Program Director may provide further indications.

# ETHICAL POLICY

Please, check the University's Ethics Code <u>here</u>. The Program Director may provide further indications.