#### card of course

|  |  |
| --- | --- |
| Subject name | Python Programming |

1. The placement of the subject in the study system

|  |  |
| --- | --- |
| 1.1. Field of study | Computer science |
| 1.2. Form and path of study | Full-time/Part-time |
| 1.3. Level of education | First-cycle studies |
| 1.4. Study profile | Practical |

|  |  |
| --- | --- |
| 1. 5. Specialty | Artificial intelligence |
| 1.6. Subject Coordinator | Dr inż. Róża Dzierżak |

2. General characteristics of the subject

|  |  |
| --- | --- |
| 2.1. Belonging to a subject group | Optional/practical |
| 2.2. Number of ECTS | 5 |
| 2.3. Language of lectures | English |
| 2.4. Semesters in which the subject is taught | III |
| 2.5.Criteria for selecting course participants | For specialization: Artificial Intelligence |

1. Learning outcomes and course delivery
	1. Subject Objectives

|  |  |
| --- | --- |
| No. | Subject Objectives |
|
| C1 | Mastering topics such as object-oriented programming, memory management, and code optimization. |
| C2 | Learning how to use popular Python tools for data analysis, application development, and task automation. |
| C3 | Creating applications and problem solutions based on real cases, preparing for work in a professional environment. |

* 1. Subject-specific learning outcomes, divided into knowledge , skills and competences , with reference to the directional learning outcomes

|  |  |  |  |
| --- | --- | --- | --- |
| No. | Description of subject learning outcomes | Reference to directional effectslearning (symbols) | Method of implementation (mark "X") |
| ST | NST |
| Classes at the University | Activities on the platform | Classes at the University | Activities on the platform |
| After passing the course, the student knows and understands **the knowledge** |
| W1 | Familiar with advanced Python programming techniques, including object-oriented programming, modules, and exception management. | INF\_W08INF\_W11 | X |  |  | X |
| W2 | Understands the principles of operation of popular libraries such as NumPy, Pandas, Matplotlib, and Flask. | X |  |  | X |
| W3 | Possesses knowledge of code testing and debugging and the tools supporting these processes. | X |  |  | X |
| W4 | Knows the basic mechanisms of working with databases in Python, including the use of libraries such as SQLite and SQLAlchemy. | X |  |  | X |
| W5 | Understands the principles of Python code optimization and is familiar with good programming practices. | X |  |  | X |
| After passing the course, the student is **able** to: |
| U1 | Can create complex programs using advanced programming techniques such as inheritance, polymorphism and exception management. | INF\_U15 INF\_U17 INF\_U19 | X |  | X |  |
| U2 | Is able to use Python libraries and frameworks to solve specific problems, e.g. data analysis, visualization, or creating web applications. | X |  | X |  |
| U3 | Can design, implement and manage databases using Python. | X |  | X |  |
| U4 | Can create unit tests and debug code using tools such as Pytest or Debugger in IDEs. | X |  | X |  |
| U5 | Can analyze and optimize code for performance and readability. | X |  | X |  |
| After completing the course, the student is ready to take part in **social competences.** |
| K1 | Is able to analyze his/her own work, consult and seek knowledge from a specialist. | INF\_K01 | X |  | X |  |
| K2 | Able to effectively communicate the results of his/her work, both in the form of documentation and technical presentation. | X |  | X |  |

3.3. Forms of teaching and their number of hours - Full-time studies (ST), Part-time studies (NST)

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Path | Lecture | Exercises | Design | Workshop | Laboratory | Seminar | Lecturer | Classes conducted using distance learning methods and techniques in the form of a lecture | Other | **ECTS points** |
| **ST** | 20 |  |  |  | 40 |  |  |  |  | 5 |
| **NST** |  |  |  |  | 20 |  |  | 10 |  | 5 |

3.4. Content of education (separately for each form of classes: (W, ĆW, PROJ, WAR, LAB, LEK, OTHER). It should be marked (X) how the given content will be implemented (classes at the university or classes on the e-learning platform conducted using distance learning methods and techniques)

TYPE OF CLASS: LECTURE

|  |  |  |  |
| --- | --- | --- | --- |
| No. | Content of the course | Reference to subject-specific learning outcomes | Method of implementation (mark "X") |
| ST | NST |
| **Classes at the University** | **Activities on the platform** | **Classes at the University** | **Activities on the platform** |
| 1. | Advanced Python Programming Techniques | W1, W3, W5 | X |  |  | X |
| 2. | NumPy and Pandas libraries | W2, W3 | X |  |  | X |
| 3. | Data visualization with Matplotlib and Seaborn . creating graphs and visual analysis. | W1, W2 | X |  |  | X |
| 4. | Flask Framework | W1, W2 | X |  |  | X |
| 5. | Working with databases in Python | W4, W5 | X |  |  | X |
| 6. | Summary of classes and discussion of grades |  | X |  |  | X |

TYPE OF CLASS: LABORATORY

|  |  |  |  |
| --- | --- | --- | --- |
| No. | Content of the course | Reference to subject-specific learning outcomes | Method of implementation (mark "X") |
| ST | NST |
| **Classes at the University** | **Activities on the platform** | **Classes at the University** | **Activities on the platform** |
| 1. | Advanced Python programming techniques : inheritance, polymorphism, exception management. | U1, K1 | X |  | X |  |
| 2. | Data analysis libraries: NumPy, Pandas , creating and processing data structures. | U2, K1 | X |  | X |  |
| 3. | Data visualization with Matplotlib and Seaborn . creating graphs and visual analysis. | U2, K1 | X |  | X |  |
| 4. | Creating web applications with Flask , basics of working with the Flask framework, creating endpoints, handling forms. | U2, U5 | X |  | X |  |
| 5. | Working with databases in Python , SQLite, SQLAlchemy, managing database connections. | U3, K1 | X |  | X |  |
| 6. | Testing and debugging code , unit testing with Pytest, debugging in IDEs. | U4, K1 | X |  | X |  |
| 7. | Optimization and good programming practices , code profiling, bottleneck identification, refactoring. | U5, K2 | X |  | X |  |
| 8. | Presentation of the final project. Review of projects, discussion, evaluation of results. | K1, K2 | X |  | X |  |

3.5. Methods of verifying learning outcomes (indication and description of methods of conducting classes and verification of achievement of learning outcomes and method of documentation)

|  |  |  |  |
| --- | --- | --- | --- |
| Subject Effects | Teaching methods | Methods of verifying learning outcomes | Documentation methods |
| KNOWLEDGE |
| W1-W5 | Lecture delivered by a specialist using tools and presentations | Passing the exam on the PUW platform. | Archived assessment on the platform. |
| SKILLS |
| U1-U4 | Practical classes performed at computer stations | Students create an application that solves a specific problem or automates a process. The project may include various functionalities, such as:File support (e.g. CSV, JSON).Data analysis using NumPy and Pandas libraries.Data visualization with Matplotlib.Simple user interface (e.g. console or web with Flask).Connecting to the database and saving results. | Archived files. |
| SOCIAL COMPETENCES |
| K1-K2 | Practical classes performed at computer stations | Students create an application that solves a specific problem or automates a process. The project may include various functionalities, such as:File support (e.g. CSV, JSON).Data analysis using NumPy and Pandas libraries.Data visualization with Matplotlib.Simple user interface (e.g. console or web with Flask).Connecting to the database and saving results. | Archived files. |

3.6. Assessment criteria for the achieved learning outcomes

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Learning effect | For a grade of 3 or "pass."the student knows and understands/is able to/is ready to | For a grade of 3.5, the student knows and understands/is able to/is ready to | For a grade of 4, the student knows and understands/is able to/is ready to | For a grade of 4.5, the student knows and understands/is able to/is ready to | For a grade of 5, the student knows and understands/is able to/is ready to |
| W | 51-60% of knowledge indicated in learning outcomes | 61-70% of knowledge indicated in learning outcomes | 71-80% of knowledge indicated in learning outcomes | 81-90% of knowledge indicated in learning outcomes | 91-100% of knowledge indicated in learning outcomes |
| U | 51-60% of skills indicated in learning outcomes | 61-70% of skills indicated in learning outcomes | 71-80% of skills indicated in learning outcomes | 81-90% of skills indicated in learning outcomes | 91-100% of skills indicated in learning outcomes |
| K | 51-60% of skills indicated in learning outcomes | 61-70% of skills indicated in learning outcomes | 71-80% of skills indicated in learning outcomes | 81-90% of skills indicated in learning outcomes | 91-100% of skills indicated in learning outcomes |

3.7. Literature

**Basic**

Lutz M., "Python. Wprowadzenie. Edycja V", Helion, 2023.

Dawson Michael., "Python dla każdego. Edycja II", Helion, 2021.

**Supplementary:**

Barry Paul, Python Rusz głową!, Helion, Gliwice, 2024

Moskała Marcin, Python od podstaw : zacznij swoją przygodę z programowaniem, Warszawa, 2023

4. Student workload - ECTS points balance

|  |  |
| --- | --- |
| **Types of student activity** | **Student Load** |
| **ST** | **NST** |
| **Classes requiring direct contact between the student and the academic teacher at the university premises** | **60** | **30** |
| Classes included in the study plan | 60 | 30 |
| **Student's own work** | **65** | **95** |
| Ongoing preparation for classes, preparation of project work/presentations/etc. | 30 | 50 |
| Preparation for passing classes | 35 | 45 |
| **TOTAL STUDENT HOURLY LOAD** | **125** | **125** |
| **Number of ECTS points** | **5** | **5** |

|  |  |
| --- | --- |
| Last change date | 30/09/2024 |
| The changes were introduced | INF Education Quality Team |
| The changes were approved | Arkadiusz Gwarda, M.A. |