**Университет ИТМО, факультет программной инженерии и компьютерной техники**

**Двухнедельная отчётная работа по «Информатике»: аннотация к статье**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Дата прошедшей лекции | Номер прошедшей лекции | Название статьи/видеолекции | Дата публикации | Размер статьи | Дата сдачи |
| 11.09.2024 | 1 | Компьютеры на основе тернарной логики перспективы их развития | 18.04.2022 | 1300 слов | 25.09.2024 |
| 25.09.2024 | 2 | СФ-блок кодирования Хаффмана для сжатия изображений по стандарту JPEG | 04.03.2022 | 1310 слов | 09.10.2024 |
| 09.10.2024 | 3 | История языка Python | 14.05.2021 | 500 слов | 23.10.202 |

Выполнил(а) Умарова А. № группы Р3118 , оценка

|  |
| --- |
| **Прямая полная ссылка на источник или сокращенная ссылка**  https://www.interestprograms.ru/article-python-istoriya-sozdaniya-yazyka-programmirovaniya |
| **Теги, ключевые слова или словосочетания**  PEP, PSW, CPython, PyPy, динамическая типизация, интерпретируемый язык |
| **Перечень фактов, упомянутых в статье**   1. Python был создан Гвидо ван Россумом в конце 1980-х - начале 1990-х годов. 2. Официальный релиз Python состоялся 20 февраля 1991 года. 3. PEP — это документы новые возможности для Python. Они играют ключевую роль в развитии языка и принятии решений сообществом. 4. Python Software Foundation - некоммерческая организация, основанная в 2001 году, которая занимается продвижением и поддержкой языка Python. Она управляет разработкой Python и поддерживает сообщество. |
| **Позитивные следствия и/или достоинства описанной в статье технологии**   1. Поддержка нескольких парадигм программирования дает разработчикам гибкость в выборе подхода к написанию кода в зависимости от задачи. 2. Легко интегрируется с другими языками для повышения производительности или использования специализированных функций. 3. Python обладает мощными библиотеками для научных вычислений и анализа данных, что делает его мощным инструментом для инженеров DS и ML. |
| **Негативные следствия и/или недостатки описанной в статье технологии**   1. Python для определенных задач медленнее компилируемых языков. 2. GIL может ограничивать параллелизм в многопоточных приложениях. 3. Динамическая типизация может привести к ошибкам, которые трудно отследить и исправить, усложнить поддержку и понимание кода. |
|  |