

Api for BCUSE

Контракт Student

Поля:

- *string* **surname** - фамилия владельца
- *string* **name** - имя владельца
- *string* **middle_name** - отчество владельца
- *string private* **login** - логин владельца
- *bytes32 private* **password** - хэш пароля владельца
- *mapping(string ⇒ uint8)* **universities** - словарь в формате {название_вуза: коэффицент_для_баллов}
- *mapping(string ⇒ string[])* **specialties** - словарь в формате {название_вуза: список_специальностей}
- *uint16* **sum_score** - сумма баллов по предметам
- *uint16[]* **score** - баллы по каждому предмету
- *string[]* **subjects** - сданные предметы

Примечание: элементы массивов score и subjects соответствуют друг другу.

Конструктор:

- *string[] memory* **en_personal_data** - массив со следующими данными:
 - 0: фамилия
 - 1: имя
 - 2: отчество (пустая строка, если нет)
 - 3: школа
 - 4: логин
 - 5: пароль

- *uint16[] memory en_score* - массив с баллами по каждому предмету
- *string[] memory en_subjects* - массив со сданными предметами

Примечание: элементы массивов `en_score` и `en_subjects` соответствуют друг другу.

Функции:

- `function set_data(string[] memory en_uni, string[][] memory en_spec, string memory en_login, string memory en_password) public {}`

Запись списка вузов в порядке приоритета и специальностей к каждому из них. На вход поступает список вузов (в порядке приоритета), двумерный список со специальностями для каждого вуза, логин, пароль.

- `function get_all_information_for_university (string memory en_uni) public view returns (string[3] memory, uint16, uint16[] memory, string[] memory, string[] memory) {}`

Возвращает информацию об ученике по заданному университету. В первой переменной лежит ФИО, потом суммарный балл, балл по каждому предмету, сданные предметы, специальности.

Контракт University

Поля:

- `mapping(string ⇒ address[]) students` - словарь в формате {название_специальности: массив_с_адресами_учеников}
- `mapping(string ⇒ uint16) amount` - словарь в формате {название_специальности: количество бюджетных мест}
- `string[] specialties` - массив с специальностями вуза

Конструктор:

- *string[] memory en_spec* - массив со специальностями
- *uint16[] memory en_amount* - массив с числом бюджетных мест

Функции:

- `function add_student (address en_student, string memory en_spec) public {}`

Функция, добавляющая студентов. На вход ей подается адрес студента и специальность, на которую он хочет поступить.

- `function get_all_students_for_speciality (string memory en_spec) public view returns(address[] memory) {}`

Возвращает список адресов студентов, поступающих на данную специальность.