

University.

Реалізуйте класс **University** в якого є:

Свойства:

- **name** - ім'я університету
- **city** - місто в якому знаходиться університет
- **zipCode** - поштовий індекс.

Методи:

- **getAddress()** - повертає інформація про університет. Ім'я, місто в форматі Uppercase і поштовий індекс. Приклад нижче.

Додати валідації:

- **name** може містити тільки з літери, пробіли і дефіси. Максимальна кількість символів: 128.
- **zipCode** може містити тільки з цифри. Максимальна кількість розрядів - 5.

Приклад:

```
const lp = new University('National University Lviv Polytechnic', 'Lviv', 79000);
console.log(lp.getAddress())
// "National University Lviv Polytechnic" - LVIV, 7900
```

User.

Реалізуйте класс **User** в якого є:

Свойства:

- **name** - ім'я користувача - **приватне**
- **surname** - прізвище користувача - **приватне**
- **bDate** - рік народження - **приватне**

Методи:

- **getFullName()** - повертає інформація про користувача, ім'я і прізвище. Приклад нижче.
- **getFullInfo()** - повертає повну інформацію про користувача, ім'я, прізвище і дата народження. Приклад нижче.

Додати валідації: **name** може містити тільки з літери і дефіси. Максимальна кількість символів: 32.

- **surname** може містити тільки з літери і дефіси. Максимальна кількість символів: 64.
- цифри. Максимальна кількість розрядів - 4. Мінімальний рік - 1900, максимальний - поточний рік (потрібно отримати динамічно за допомогою класу Date)

Приклад:

- **bDate** може містити тільки з

```
const user1 = new User('Misha', 'Savchak', 1991);
console.log(user1.getFullName());
// Misha Savchak
console.log(user1.getFullInfo());
// Misha Savchak, 1991
```

Student.

Реалізуйте клас **Student** що наслідується від класу **User** в якого є:

Свойства:

- **counter** - містить інформацію про кількість створених об'єктів Student - **статична**
- **year** - містить інформацію про дату вступу до університету - **приватна**
- **university** - містить всю інформацію про університет. Параметр не обов'язковий - **приватна**

Методи:

- **getCourse()** - повертає інформація про курс від 1-6. Курс обчислюється так: потрібно від поточного року відняти рік вступу до університету. Поточний рік отримуєте самостійно. Приклад нижче.
- **isFinished()** - повертає true або false в залежності від того чи студент закінчив виш. Обчислюється так: **Поточний рік - Рік вступу**. Якщо результат обчислення більше 6 - ТАК, менше - НІ. Приклад нижче.
- **getCounter()** - статичний метод. Повертає кількість створених об'єктів User. Приклад нижче.
- **getFullInfo()** - повертає повну інформацію про користувача з батьківського класу і додає ім'я університету і рік вступу. Приклад нижче.

Додати валідації:

- **counter** початкове значення 0.
- **year** може містити тільки з цифри. Максимальна кількість розрядів - 4. Мінімальний рік - 1900, максимальний - поточний рік (потрібно отримати динамічно за допомогою класу Date)
- **university** - додати перевірку чи змінна належить класу (typeof).

```
const lp = new University('National University Lviv Polytechnic', 'Lviv', 79000);
const student1 = new Student('Misha', 'Savchak', 1991, 2014, lp );
console.log(student1.getCourse()); // 6
console.log(student1.isFinished()); // true
console.log(Student.getCounter()); // 1
console.log(student1.getFullInfo());
// Misha Savchak, 1991, "National University Lviv Polytechnic", 2014

const student2 = new Student('FirstName', 'LastName', 1991, 2019);
console.log(student2.getCourse()); // 2
console.log(student2.isFinished()); // false
console.log(Student.getCounter()); // 2
console.log(student2.getFullName());
// FirstName LastName
console.log(student2.getFullInfo());
// FirstName LastName, 1991, Unknown University
```

Worker.

Реалізуйте клас **Worker** що наслідується від класу **User** і в якого є:

Свойства:

- **counter** - містить інформацію про кількість створених об'єктів Student - **статична**
- **rate** - ставка за день роботи.
- **days** - містить інформацію про дату вступу до університету - **приватна**
- **retired** - true або false, містить інформацію чи людина звільнена. За замовчуванням - false - **приватна**

Методи:

- **isRetired()** - повертає true або false. Приклад нижче.
- **getSalary()** виводить зарплату робітника. Зарплата - це days * rate. Якщо робітник звільнений виводить 0. Приклад нижче.

Додати валідації:

- **counter** початкове значення 0;
- **rate** може містити тільки цифри;
- **days** може містити тільки цифри;

Приклад:

```
const worker = new Worker('Misha', 'Savchak', 1991, 20, 34);
console.log(worker.getFullName()); // Misha Savchak
console.log(worker.isRetired()); // false
console.log(worker.getSalary()); // 680
console.log(worker.getDays()); // 34
console.log(worker.getRate()); // 20
worker.setRetired(true);
console.log(worker.getSalary()); // 0
```

Замітки:

- Для форматування тексту використовувати Шаблонні літерали.
- Імена приватні властивостей повинні називатися з символу підчеркування.
this.name -> this._name
- Додати коментарі до класів і методів
- Константи
- Всі свойства класу мають мати функції сетери і гетери
getName() і setName('UserName')
getSurname() і setSurname('Surname')
- Клас **University** має бути в окремому файлі і експортований за замовчуванням
- Клас **User** має бути в окремому файлі і експортований за замовчуванням
- Клас **Student** має бути в окремому файлі і експортований за замовчуванням
- Клас **Worker** має бути в окремому файлі і експортований за замовчуванням

Створити 3 екземпляри класу University, 1 - User, 2 - Students, і 3 Workers. Навести свої приклади виконання.