



Python. Подготовка к собеседованию

Урок 2

Парадигма ООП особенности и отличия от других ЯП

Сведения об ООП. Словари. Инкапсуляция, наследование, полиморфизм. Преимущества Python.

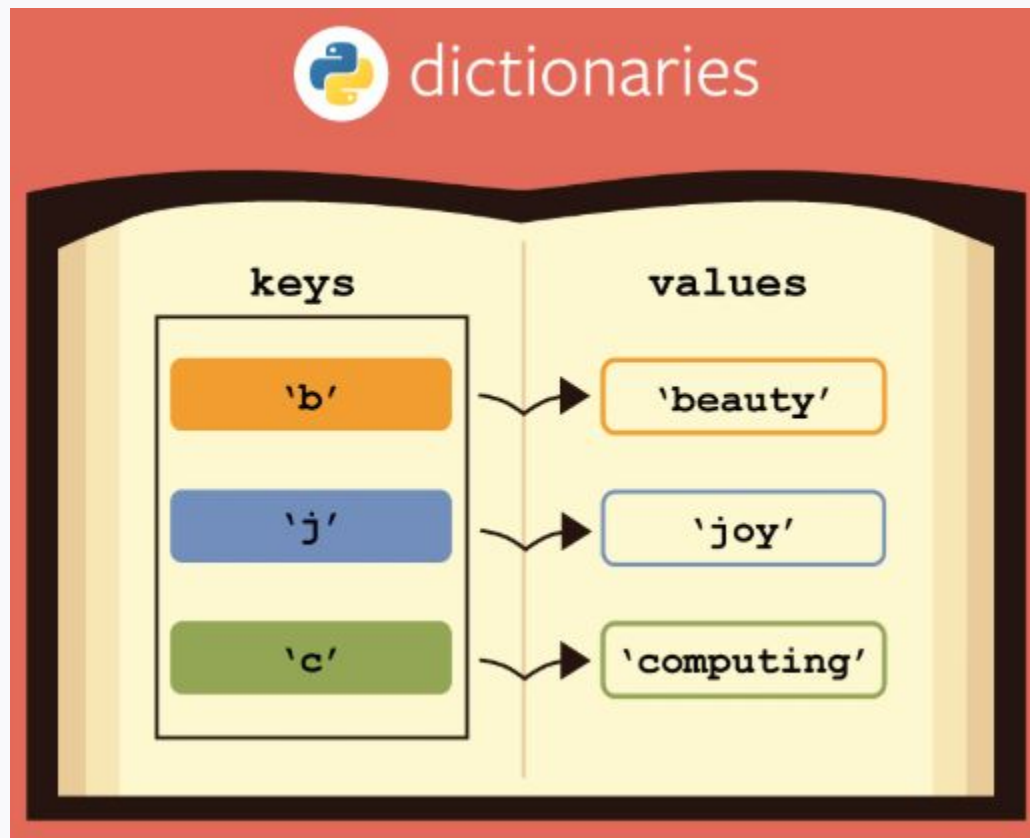
Цели урока

Повторить словари. Разобрать концепцию ООП

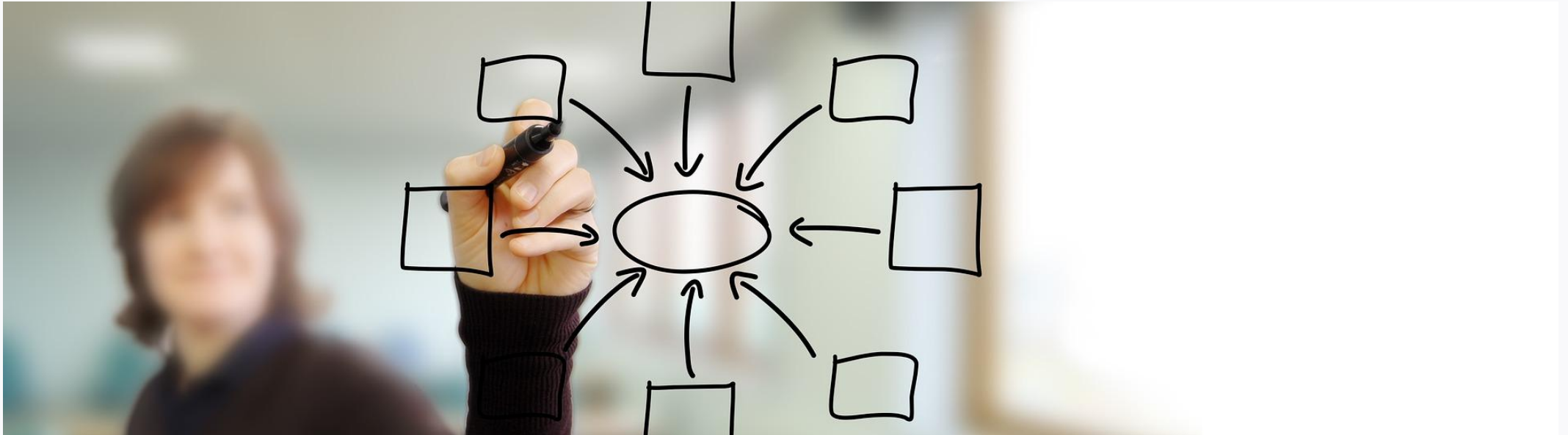
- Словари;
- ООП;
- Основные свойства ООП;
- Отличия.



Словарь как структура данных



ООП



Классы

Класс — это пользовательский тип, состоящий из методов и атрибутов.



Основные свойства ООП



- Инкапсуляция
- Наследование
- Полиморфизм



Инкапсуляция

Инкапсуляция - ограничение доступа к составляющим объект компонентам (методам и переменным). Инкапсуляция делает некоторые из компонентов доступными только внутри класса.



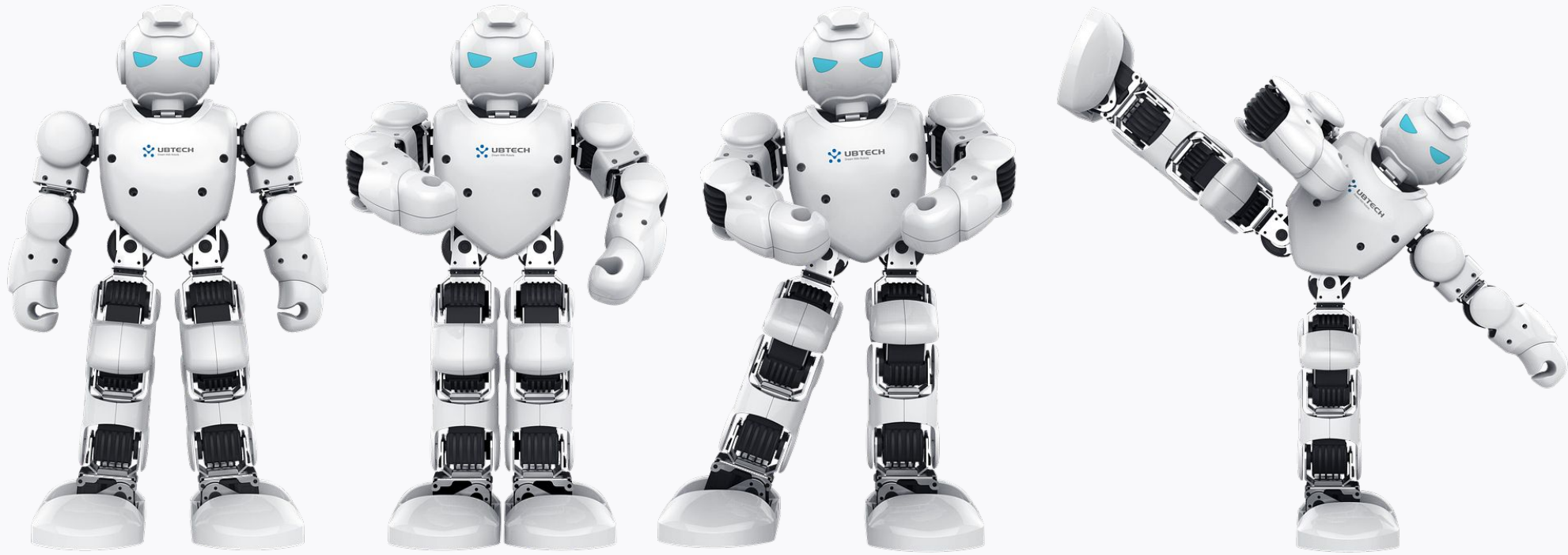
Наследование

Наследование позволяет создавать специализированные классы на основе базовых. Это дает возможность не писать повторный код.



Полиморфизм

Полиморфизм заключается в том, что в разных объектах одна и та же операция может выполнять различные функции. Слово «полиморфизм» имеет греческого происхождения и означает «имеющий многие формы».



Отличия Python от других языков программирования



Практическое задание



Практическое задание (часть 1)

1. Проверить механизм наследования в Python. Для этого создать два класса. Первый — родительский (**ItemDiscount**), должен содержать статическую информацию о товаре: название и цену. Второй — дочерний (**ItemDiscountReport**), должен содержать функцию (**get_parent_data**), отвечающую за отображение информации о товаре в одной строке. Проверить работу программы, создав экземпляр (объект) родительского класса.
2. Инкапсулировать оба параметра (название и цену) товара родительского класса. Убедиться, что при сохранении текущей логики работы программы будет сгенерирована ошибка выполнения. Усовершенствовать родительский класс таким образом, чтобы получить доступ к защищенным переменным. Результат выполнения заданий 1 и 2 должен быть идентичным.
3. Реализовать возможность переустановки значения цены товара. Необходимо, чтобы и родительский, и дочерний классы получили новое значение цены. Следует проверить это, вызвав соответствующий метод родительского класса и функцию дочернего (функция, отвечающая за отображение информации о товаре в одной строке).



Практическое задание (часть 2)

4. Реализовать расчет цены товара со скидкой. Величина скидки должна передаваться в качестве аргумента в дочерний класс. Выполнить перегрузку методов конструктора дочернего класса (метод `__init__`, в который должна передаваться переменная — скидка), и перегрузку метода `__str__` дочернего класса. В этом методе должна пересчитываться цена и возвращаться результат — цена товара со скидкой. Чтобы все работало корректно, не забудьте инициализировать дочерний и родительский классы (вторая и третья строка после объявления дочернего класса).
5. Проверить на практике возможности полиморфизма. Необходимо разделить дочерний класс **ItemDiscountReport** на два класса. Инициализировать классы необязательно. Внутри каждого поместить функцию **get_info**, которая в первом классе будет отвечать за вывод названия товара, а вторая — его цены. Далее реализовать выполнение каждой из функции тремя способами.



Дополнительные материалы

1. <https://habr.com/post/370831/>.
2. <https://habr.com/company/cit/blog/262887/>.
3. <http://www.quizful.net/category/python>.
4. <https://habr.com/post/49671/>.
5. <http://kip-world.ru/kakie-byvayut-tipy-dannyh-v-python.html>.

