

Міністерство освіти і науки України

Національний технічний університет України

«Київський політехнічний інститут

ім. І.І.Сікорського»

Навчально-науковий комплекс

«Інститут прикладного системного аналізу»

**Лабораторна робота №5**

з курсу "Проектування інформаційних систем"

на тему

«Система автоматичного створення документації»

Виконав:

студент ІV курсу

групи ДА-41

Клещ Кирило

Варіант-9

Київ 2017

**Мета роботи:**

За допомогою системи генерації довідника користувача створитидокумент у форматі PDF і HTML для архітектурної програмної моделі.

**Задача:**

1. Вивчити теги системи генерації керівництва користувача.
2. Створити опис для всіх класів API з описом призначення кожного класу, методів класу і членів класу.
3. Згенерувати документацію у форматах PDF, HTML.

Короткі теоретичні відомості

Генератор документації є програмний інструмент, який генерує документацію з програмного забезпечення призначена для програмістів (API документацію) або кінцевим користувачам (End-User Guide), або обидва з набору спеціально прокоментував файли вихідного коду, а в деяких випадках, бінарні файли. Документ генерації можна розділити на кілька типів документів:

* Пакетні документи (все автоматизовано документи)
* Інтерактивні документи (документи, які не можуть бути отримані автоматично)

**Завдання:**

1. Для кожного з класів API в код програми додати теги з описом керівництва користувача для архітектурної програмної моделі.
2. Обрати Coding Convention. Оформити код програми згідно Coding Convention.
3. Встановити налаштування системи автоматичного створення керівництва користувача.
4. Згенерувати HTLM документацію керівництва користувача.
5. Згенерувати PDF документацію керівництва користувача.

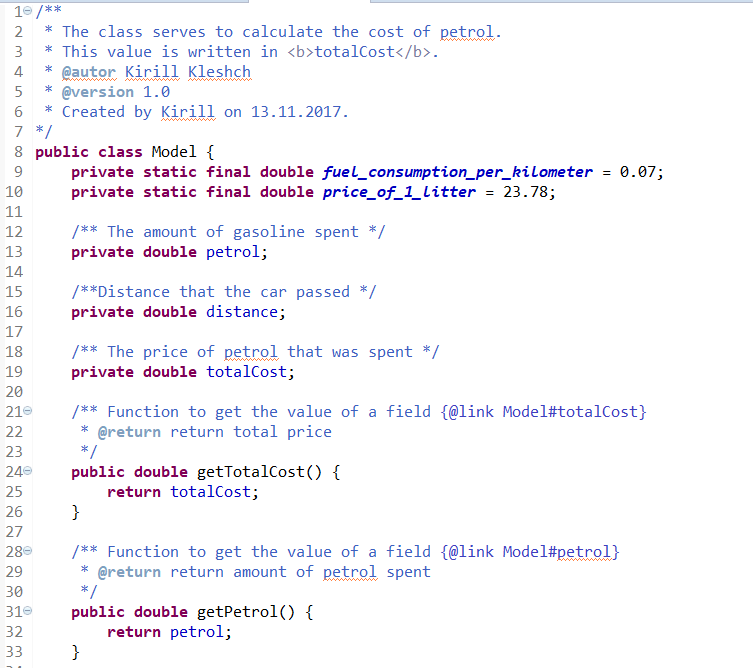
**Хід роботи:**

**Javadoc** — стандартный генератор документации в HTML-формате из комментариев исходного кода.  
  
Для создания описания к элементу(поле, класс, метод) используются специальный комментарий, расположенный выше этого элемента:

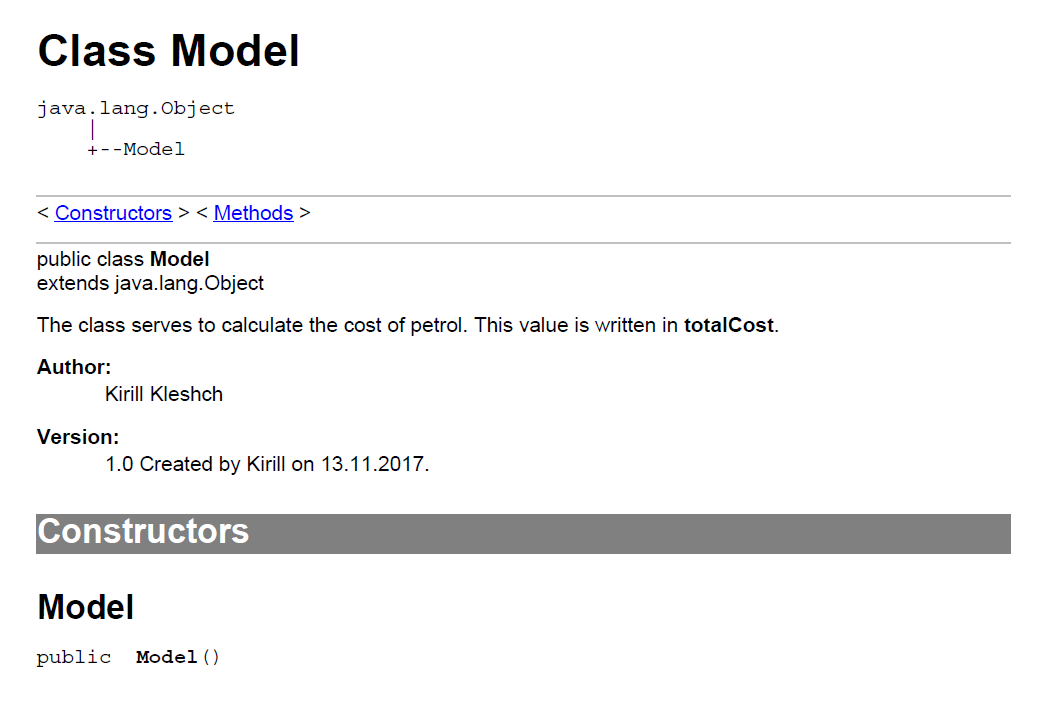
/\*\* Описание \*/

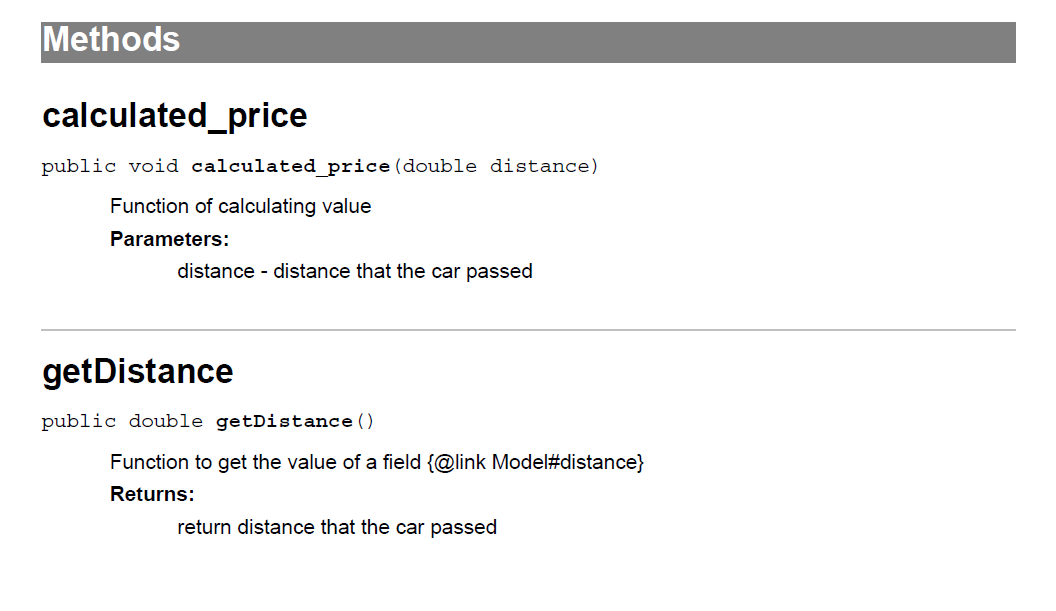
Для документирования можно использовать дескрипторы, вот некоторые из них:  
**@author** — автор  
**@version** — версия  
**@since** — указывает с какой версии появился этот блок кода  
**@see** — ссылка на другое место в документации  
**@param** — передаваемый параметр методу  
**@return** — описание возвращаемого значения метода  
**@exception** и **@throws** — описание исключений  
**@deprecated** — документирование устаревших частей кода  
**{@link}** — создание ссылки, можно вставлять в любое место  
**{@value}** — описание значения переменной

**Приклад документованого коду:**

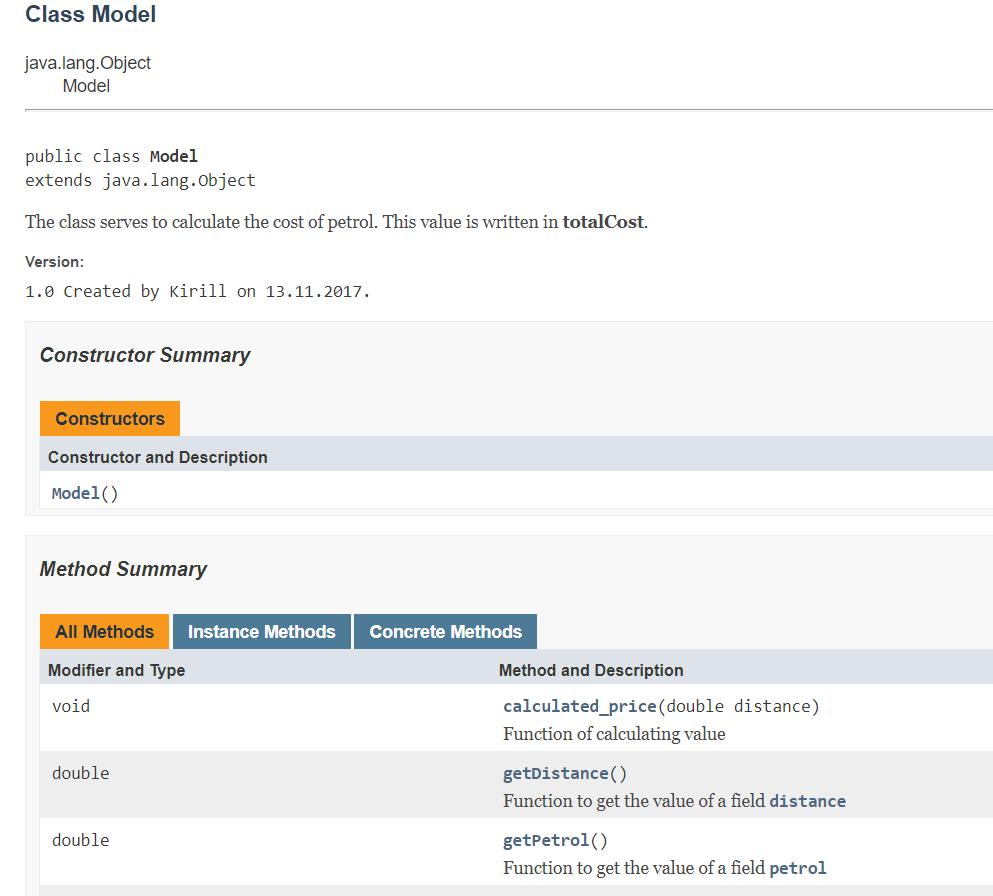


**Документація у форматі PDF:**





**Документація у форматі HTML:**





**Висновки:**

В ході лабораторної роботи, була згенерована документація за допомгою фреймворка JavaDoc. Було використано середовище розробки Eclipse. На виході було отримано документацію для классу Model у форматі HTML та PDF.