**Задание 3.**

Предполагается, что перед выполнением задания 3 были выполнены задание 1 (или хотя бы установлен Microsoft SQL Server Management Studio и в него были загружены данные) и задание 2 (хотя бы был запущен скрипт в SQL). Если этого не сделано, нужно выполнить сначала предыдущие задания (или хотя бы упомянутые его части).

**Необходимый софт:**

* **Python** (Ссылка на скачивание: <https://www.python.org/downloads/windows/> ). В ходе работы будут использоваться juriter notebook.
* **Microsoft SQL Server Management Studio** (откуда скачать и как установить описано в инструкции к заданию 1)
* Драйвер**pyodbc (**Ссылка на скачивание и на инструкцию по установке: <https://docs.microsoft.com/ru-ru/sql/connect/python/pyodbc/step-1-configure-development-environment-for-pyodbc-python-development?view=sql-server-2017>).

**Имеющиеся данные:**

* Исходные данные к заданиям, загруженные в СУБД в базу данных HW. В первую очередь, таблица с результатами торгов, на основе которой будет создана таблица с интересующими нас доходностями.
* В ходе второго задания в БД HW была создана вспомогательная таблица bond\_information, которая будет необходима для создания структуры платежей. Таблица находится в БД HW.
* Данные ЦБ о безрисковой доходности: <https://www.cbr.ru/hd_base/zcyc_params/>). Необходимо скачать ежедневные данные с 2005 по 2015 гг в файл xlsx с названием «RiskFree» и пометить данный файл в рабочую директорию python.

Предполагается, что был связан с Microsoft SQL Server Management Studio с помощью pyodbc. Был создан новый пользователь с проверкой подлинности SQL server и проделаны остальные действия, чтобы можно было делать запросы к БД из python. Если нет, то нужно их проделать, руководствуясь инструкцией к заданию 2).

**Последовательность выполнения задания:**

1. Скачать и установить все упомянутые программы (пользоваться инструкциями к заданиям 1,2). Удостовериться, что juriter notebook работают. Выполнить хотя бы первый пункт первого задания, после чего запустить SQL скрипт ко второму заданию. Удостовериться, что был создан новый пользователь с проверкой подлинности SQL server.
2. Убедиться, что все данные загружены необходимые данные есть и загружены.
3. Запустить скрипт в SQL server. Скрипт должен создать на основе таблицы с результатами торгов вспомогательную таблицу с доходностями бид и аск к погашению и оферте с числовым форматом данных.
4. Зайти в juriter notebook. Запустить код к заданию 3. Загрузить файл с кривой бескупонной доходностью. Соединить python с СУБД.
5. Далее последовательно запускать команды кода, пока не будут выполнены все задания.

**Ожидаемые результаты:**

* В SQL server должна быть создана таблица с доходностями бид и аск к погашению и оферте.
* Будет загружена и выведена на экран таблица с ежедневными данными по безрисковой доходности.
* Пользователь введет ISIN интересующей его облигации.
* В juriter notebook выполнится команда, подключающая соединение sql server и python. Из БД будет загружена таблица с бид и аск доходностями к погашению и оферте с указанием дат по выбранной облигации.
* Будет построена структура платежей по выбранной облигации.
* На каждую дату торгов будет посчитана безрисковая стоимость облигации.
* На основе безрисковой стоимости облигации и структуры платежей на каждую дату торгов будет посчитана безрисковая ytm по интересующей ценной бумаге.
* Будет создан график, отражающий динамику спрэдов обычных и безрисковой доходностей.
* По спрэдам будут построены гистограммы.