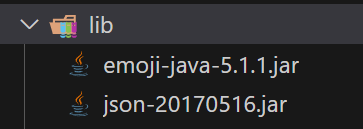
**Lesson 26. EmojiBot**

Після знайомства із тим, як почати працювати із телеграм-ботом, пропонуємо навчитись додавати у повідомлення бота емоджі, які давно вже стали невд’ємною частиною спілкування онлайн.

Емоджі дозволять відповіді бота зробити «живішими» і приємнішими для користувача.

**Emoji**

Для використання емоджі у нашому боті знадобляться ще додаткові бібліотеки:

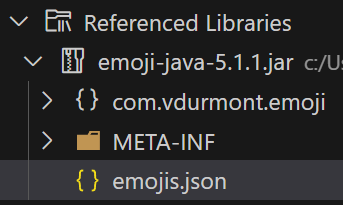
Далі на прикладі простого бота з’ясуємо як додати до повідомлення емоджі. Перш ніж це зробити варто заглянути в клас Emoji з відповідної бібліотеки, аби з’ясувати з яких з чого складається об’єкт емоджі.

Нас найбільше будуть цікавити ось ці три властивості:

**aliases** – «псеводніми» емоджі, щось наче кодові рядки, за якими емоджі можна знайти;

**tags** – за тегами емоджі можуть бути роздідлені на категорії, або тег може містити якусь допоміжну інформацію про емоджі;

**unicode** – код символу емоджі (саме він повинен бути в тексті повідомлення, що телеграм відобразив його як картинку);



Детальну інформацію про емоджі можна знайти у файлі із бібліотеки:

Також з’ясувати еліас емоджі можемо в самому Telegram навівши на якийсь із них курсором:

Еліас зазвичай починається із закінчується символом «**:**». Знаючи еліас можемо відтворити юнікод емоджі у повідомленні. Для цього можна скористатись одним із двох запропонованих способів:

1. Клас **EmojiParser**;

Потрібний нам метод **parseToUnicode**() приймає у якості параметру рядок тексту, а повертає такий самий рядок, але всі знайдені еліаси в ньому заміщує юнікодом емоджі.

1. Клас **EmojManager**;

Даний клас дозволяє за допомогою методу **getForAlias**() знайти об’єкт емоджі за його еліасом. Далі з отриманого об’єкту можна отримати про нього (про емоджі) всі необхідні дані, втому числі юнікод, який потім додати у текстовий рядок повідомлення.

У якості прикладу та поля для досліджень пропонуємо відтворити приклад на скріншоті нижче. В рамках телеграм бота можна поекпериментувати із еліасами, емоджі та запропонованими методами:



Дані класи мають ще багато іншого функціоналу, який ми в рамках курсу не будемо використовувати. Однак це не значить, що самостійно в нього заглибитись ви (учні) не можете за допомогою функціоналу автозаповнення та перегляду файлів бібліотек.

**Practice Tasks**

**Завдання 1**. Створіть бота, що відповідатиме на будь-яке текстове повідомлення тим самим повідомленням.

Усе що нам необхідно змінити у попередньому боті, так це метод *onUpdateReceived*. Тепер він повинен надсилати відповідь не просто в консоль, а в чат звідки було отримано апдейт.

**Homework**

**Завдання 1.** Повіторіть вдома код найпростішого телеграм-бота, або ж Mirror Bot’а. Переконайтесь, що все працює правильно.