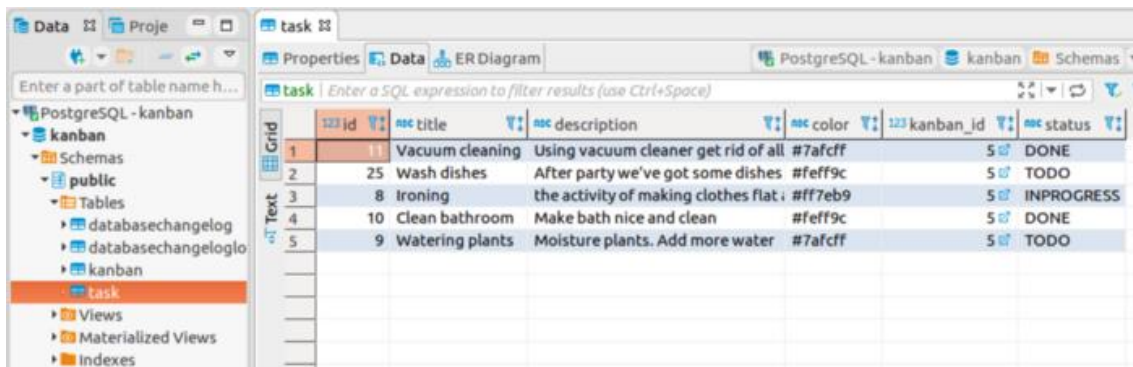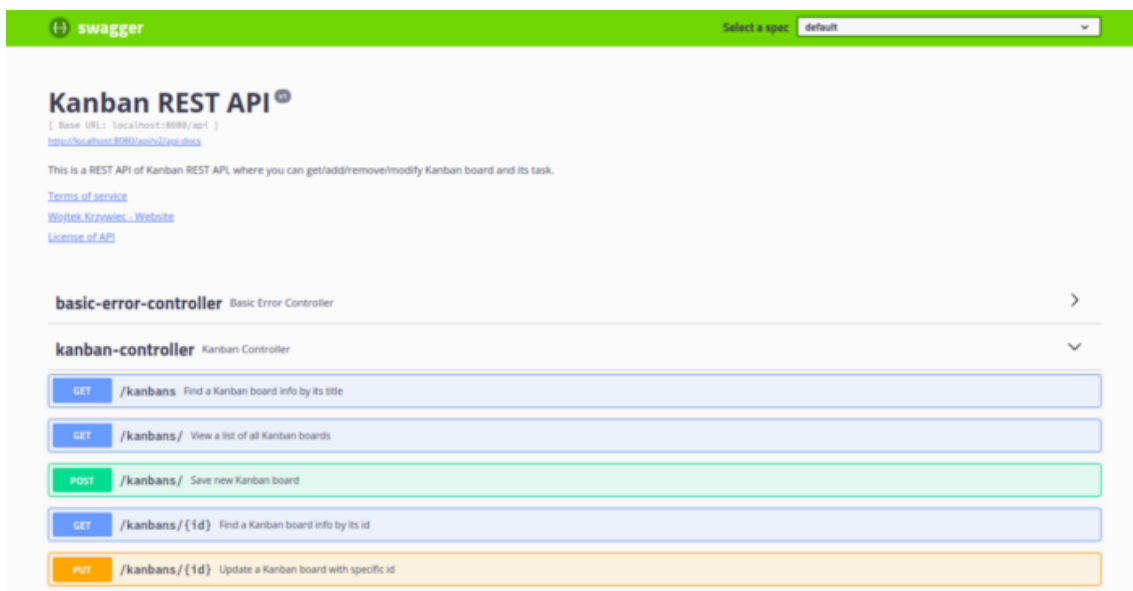# Assignment

Step 1 : Install docker compose (Install docker as well if not already installed)

Step 2 : Create a simple docker compose file with 2 services (preferably a backend and DB service)

Step 3: Run the docker-compose up -d command

Step 4: Finally run docker-compose ps command to show the list of services stood up by docker compose

Step 5: Upload the docker compose file along with the screenshot of docker-compose ps command output

```
PS C:\Users\PC\Downloads\kanban-board-master> docker-compose up -d
Creating network "kanban-board-master_default" with the default driver
Creating volume "kanban-board-master_kanban-data" with default driver
Pulling kanban-postgres (postgres:9.6-alpine)...
9.6-alpine: Pulling from library/postgres
59bf1c3509f3: Already exists
c50e01d57241: Pull complete
a0646b0f1ead: Pull complete
912227b294ee: Pull complete
696b75eaa88e: Pull complete
0d83746bb80b: Pull complete
3a431a2253cd: Pull complete
24ff05c04760: Pull complete
587cf5e11ca1: Pull complete
Digest: sha256:8342bcb43446694428ec6594e72e4299692854f0fc3aca090b0ab46f4c7f32a1
Status: Downloaded newer image for postgres:9.6-alpine
Building kanban-app
[+] Building 231.6s (16/16) FINISHED
```

```
[+] Building 231.6s (16/16) FINISHED
 => [internal] load build definition from Dockerfile                                                      0.0s
 => => transferring dockerfile: 321B                                                                      0.0s
 => [internal] load .dockerignore                                                                         0.0s
 => => transferring context: 2B                                                                           0.0s
 => [internal] load metadata for docker.io/library/openjdk:8-alpine                                       2.0s
 => [internal] load metadata for docker.io/library/maven:3.6.1-jdk-8-slim                                 2.0s
 => [auth] library/openjdk:pull token for registry-1.docker.io                                            0.0s
 => [auth] library/maven:pull token for registry-1.docker.io                                              0.0s
 => [stage-1 1/2] FROM docker.io/library/openjdk:8-alpine@sha256:94792824df2df33402f201713f932b58cb9de94a0cd5241  45.7s
 => => resolve docker.io/library/openjdk:8-alpine@sha256:94792824df2df33402f201713f932b58cb9de94a0cd524164a0f2283  0.0s
 => => sha256:44b3cea369c947527e266275cee85c71a81f20fc5076f6ebb5a13f19015dce71 947B / 947B                0.0s
 => => sha256:a3562aa0b991a80cfe8172847c8be6dbf6e46340b759c2b782f8b8be45342717 3.40kB / 3.40kB            0.0s
 => => sha256:94792824df2df33402f201713f932b58cb9de94a0cd524164a0f2283343547b3 1.64kB / 1.64kB            0.0s
 => => sha256:e7c96db7181be991f19a9fb6975cdbbd73c65f4a2681348e63a141a2192a5f10 2.76MB / 2.76MB           14.5s
 => => sha256:f910a506b6cb1dbec766725d70356f695ae2bf2bea6224dbe8c7c6ad4f3664a2 238B / 238B               12.7s
 => => sha256:c2274a1a0e2786ee9101b88f76111f9ab8019e368dce1e325d3c284a0ca33397 70.73MB / 70.73MB         42.6s
 => => extracting sha256:e7c96db7181be991f19a9fb6975cdbbd73c65f4a2681348e63a141a2192a5f10                 0.2s
 => => extracting sha256:f910a506b6cb1dbec766725d70356f695ae2bf2bea6224dbe8c7c6ad4f3664a2                 0.0s
 => => extracting sha256:c2274a1a0e2786ee9101b88f76111f9ab8019e368dce1e325d3c284a0ca33397                 1.7s
 => [build 1/6] FROM docker.io/library/maven:3.6.1-jdk-8-slim@sha256:f7ffbcfeae6e5d56019afeb8c4b288afe73d25b4a0b  51.4s
 => => resolve docker.io/library/maven:3.6.1-jdk-8-slim@sha256:f7ffbcfeae6e5d56019afeb8c4b288afe73d25b4a0b0b2954f  0.0s
 => => sha256:96a712cf92114bfb8c8322f9d705e7bccc415623e3243410209b21c7d66540cf 9.28kB / 9.28kB            0.0s
 => => sha256:1ab2bdfe97783562315f98f94c0769b1897a05f7b0395ca1520ebee08666703b 27.09MB / 27.09MB         10.1s
 => => sha256:7aaf9a088d61d78b50cd2a58984cc89ed9c0c5c66124fd74b4baa0cba32524c1 3.25MB / 3.25MB            2.3s
 => => sha256:80a55c9c9fe89b163322e46d017f346fab100604a4a0a63a1a790b89c1c2105f 210B / 210B               0.6s
 => => sha256:f7ffbcfeae6e5d56019afeb8c4b288afe73d25b4a0b0b2954fe5866143c1dda0 320B / 320B                0.0s
 => => sha256:808962fc361cf04966f559f26a195d567a900ed9a59ddc3a24ad552a7ed1e8e8 2.00kB / 2.00kB           0.0s
 => => sha256:9110ea0d47ee3b75e48971381c280070f0d76603288b1da30af905152b7d8250 104.45MB / 104.45MB       45.0s
 => => sha256:05d5ec3a150f8ce48f66684ecac1ddf0085c48064f695f0bb0f901b99572928c 2.61MB / 2.61MB            4.7s
 => => sha256:21c130b36268accd61d82bb71cc24b5bccff4607c0a432e3ce9bef4889d137b3 9.16MB / 9.16MB           12.3s
 => => sha256:3a083b0b9451be9236738d85f518aaafd98714e14f05ed395f2e3e3cc25f80ea 854B / 854B               10.4s
 => => extracting sha256:1ab2bdfe97783562315f98f94c0769b1897a05f7b0395ca1520ebee08666703b                 1.5s
 => => sha256:125924f33b34ab35df12edff4a5d6c35588c2395dd77745636d5d85c75dc3b21 356B / 356B               10.7s
 => => extracting sha256:7aaf9a088d61d78b50cd2a58984cc89ed9c0c5c66124fd74b4baa0cba32524c1                 0.2s
 => => extracting sha256:80a55c9c9fe89b163322e46d017f346fab100604a4a0a63a1a790b89c1c2105f                 0.0s
 => => extracting sha256:9110ea0d47ee3b75e48971381c280070f0d76603288b1da30af905152b7d8250                 2.6s
 => => extracting sha256:05d5ec3a150f8ce48f66684ecac1ddf0085c48064f695f0bb0f901b99572928c                 0.2s
 => => extracting sha256:21c130b36268accd61d82bb71cc24b5bccff4607c0a432e3ce9bef4889d137b3                 0.2s
 => => extracting sha256:3a083b0b9451be9236738d85f518aaafd98714e14f05ed395f2e3e3cc25f80ea                 0.0s
 => => extracting sha256:125924f33b34ab35df12edff4a5d6c35588c2395dd77745636d5d85c75dc3b21                 0.0s
```