

Spring Data Exam – 04 December 2020

Instagraph – Lite

1. Functionality Overview

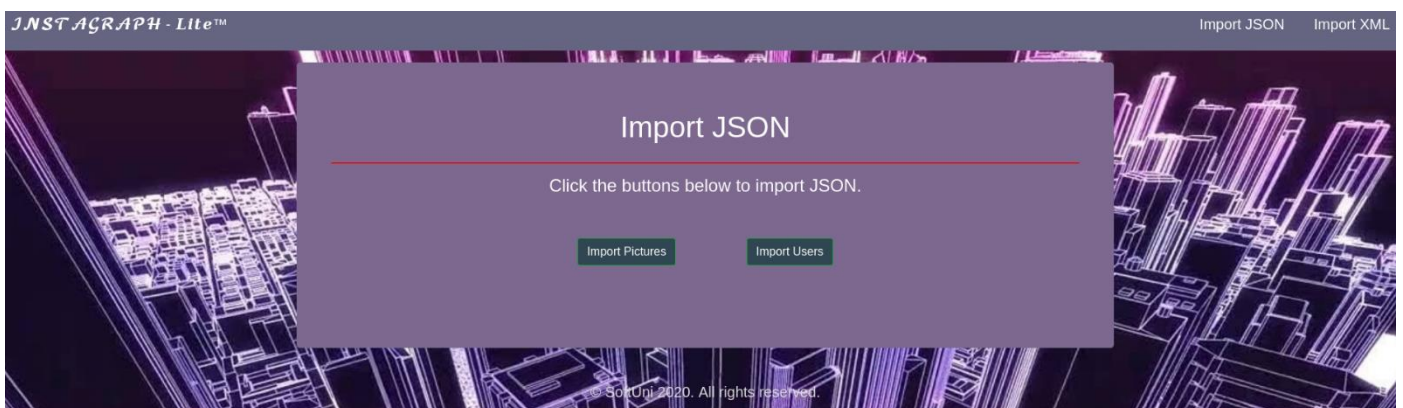
The application should be able to easily **import** hard-formatted data and **support functionality** for also **exporting** the imported data. The application is called – **Instagraph - Lite**.

Look at the pictures below to see what must happen:

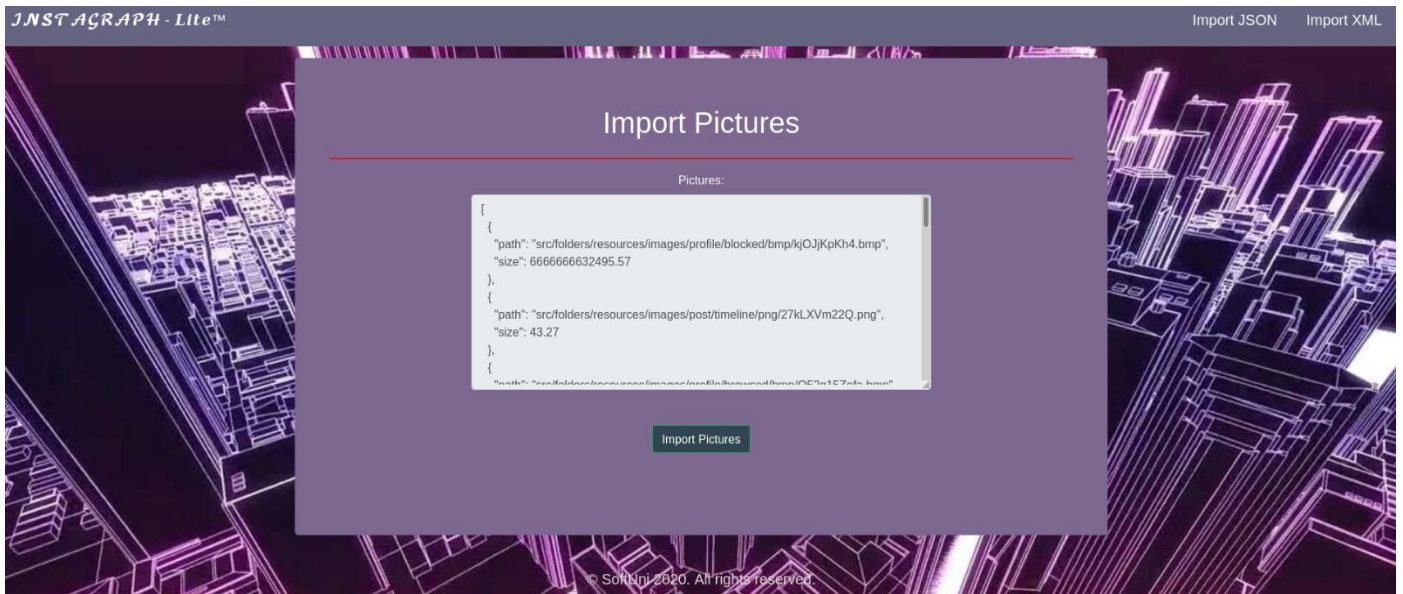
- Home page before importing anything:



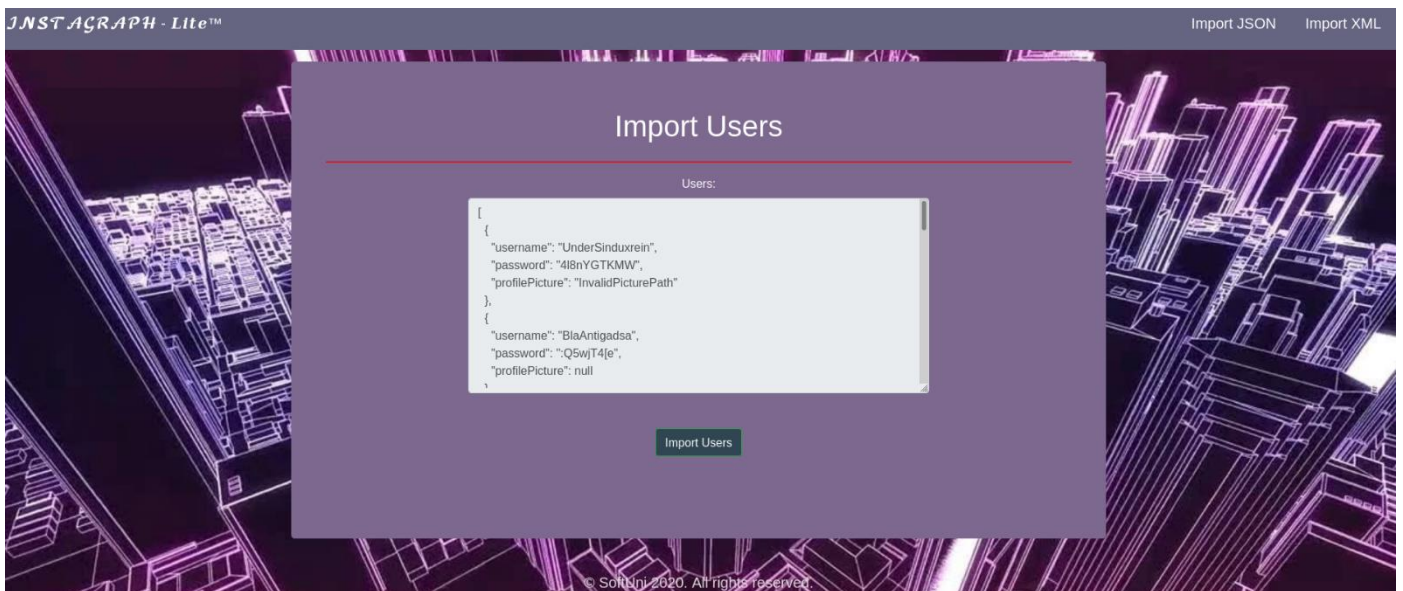
- Import JSON page before importing anything:



- Import Pictures first:



- Import Users:



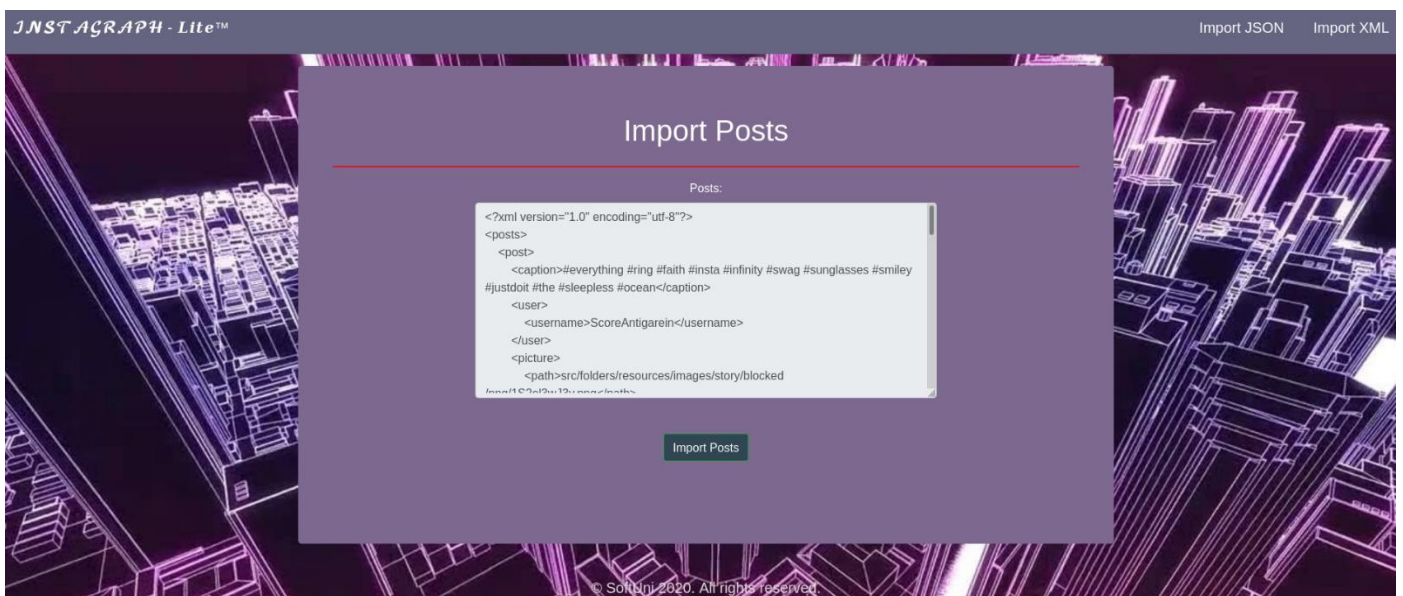
- Import JSON page after importing both files:



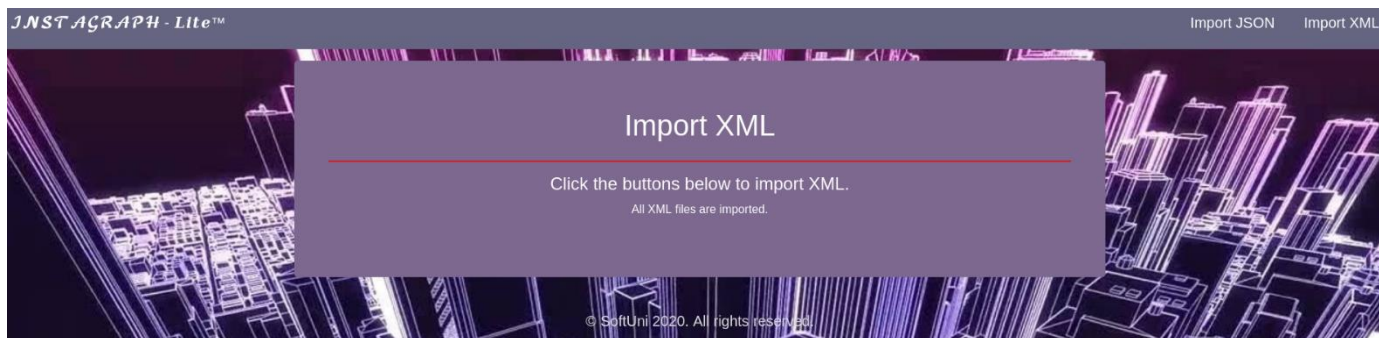
- Import XML page before importing the given data:



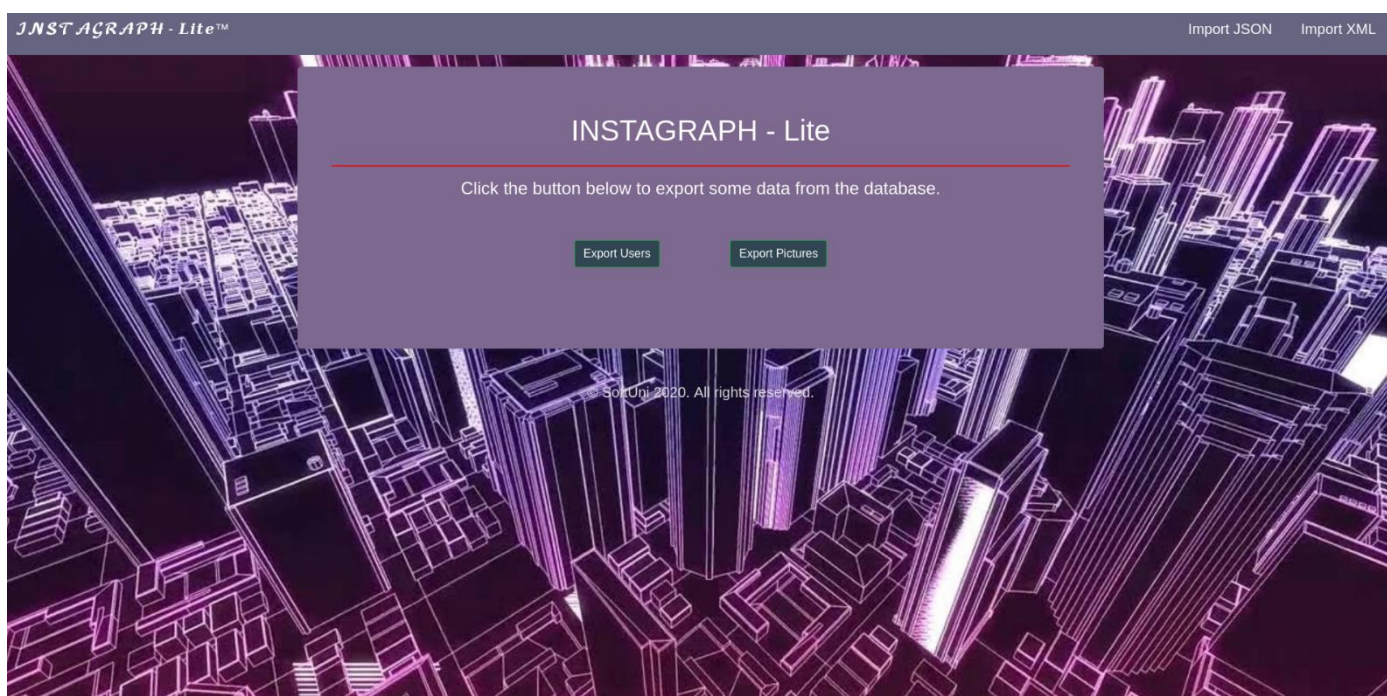
- Import Posts:



- Import XML page after importing the data:



- Home page after the data is imported:



2. Project Skeleton Overview

You will be given a **Skeleton**, containing a **certain architecture (MVC)** with **several classes**, some of which – completely empty. The **Skeleton** will include the **files** with which you will **seed** the **database**.

3. Model Definition

There are 3 main models that the **Instagraph database** application should contain in its functionality.

Design them in the **most appropriate** way, considering the following **data constraints**:

Pictures

- **id** – integer number, **primary identification field**, auto increment.
- **path** – a char sequence. **Cannot be null**. The **path** is **unique**.
- **size** – a floating point number. **Cannot be null**. Must be between **500** and **60000** (both numbers are **INCLUSIVE**)

Users

- **id** – integer number, **primary identification field**, auto increment.
- **username** – a char sequence. **Cannot** be **null**. The **username** is **unique**. Must be between **2 and 18** (both numbers are **INCLUSIVE**)
- **password** – a char sequence. **Cannot** be **null**. Must be at least **4 characters long**, **inclusive**.
- **profilePicture** – a **Picture**. **Cannot** be **null**.

Posts

- **id** – integer number, **primary identification field**, auto increment.
- **caption** – a char sequence. **Cannot** be **null**. Must be **at least 21 characters**, **inclusive**.
- **user** – a **User**. **Cannot** be **null**.
- **picture** – a **Picture**. **Cannot** be **null**.

NOTE: Name the entities and their class members, **exactly** in the **format stated** above. Do not name them in snake case with the dashes, of course.

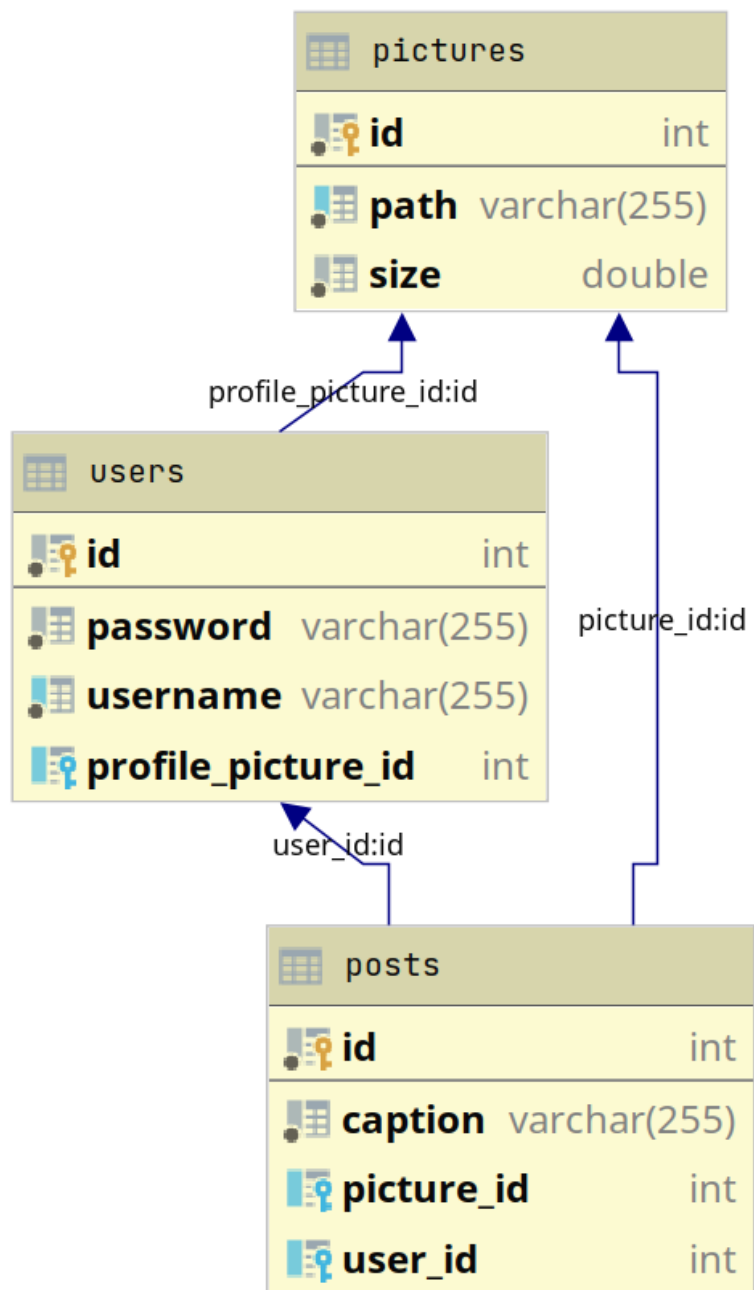
Relationships

Your partners gave you a little hint about the more complex relationships in the database, so that you can implement it correctly.

One **Picture** may have many **Users**, but one **User** may have only one **Picture**.

One **User** may have many **Posts**, but one **Post** may be from only one **User**.

One **Post** may have only one **Picture**, but one **Picture** can be in many **Posts**.



4. Data Import

Use the provided files to populate the database with data. Import all the information from those files into the database.

You are not allowed to modify the provided files.

ANY INCORRECT data should be **ignored** and a message should be printed.

- **Picture: Invalid Picture**
- **User: Invalid User**
 - If the **picture** is not in the database, do not add the user to the database and return "Invalid User"
- **Post: Invalid Post**

- If the **picture or user is not in the database**, do not add the post to the database and return **"Invalid Post"**

When the import is finished

- **Picture: Successfully imported Picture, with size {size}**
 - Format the picture size to the 2nd digit after the floating point.
- **User: Successfully imported User: {username}**
- **Post: Successfully imported Post, made by {username}**

JSON Import

Pictures (Pictures.json) pictures.json
<pre>[{ "path": "src/folders/resources/images/profile/blocked/bmp/kj0JjKpKh4.bmp", "size": 632495.57 }, { "path": "src/folders/resources/images/post/timeline/png/27kLXVm22Q.png", "size": 43.27 }, { "path": "src/folders/resources/images/profile/browsed/bmp/Q52q15Zefa.bmp", "size": null }, { "path": null, "size": 6638.98 }, { "path": "src/folders/resources/images/post/timeline/png/27kLXVm22Q.png", "size": 44273.27 }, { "path": "src/folders/resources/images/uploads/blocked/png/45Q92hqN5V.png", "size": 51087.72 }, ...]</pre>

Invalid Picture
Invalid Picture
Invalid Picture
Invalid Picture
Successfully imported Picture, with size 44273.27
Successfully imported Picture, with size 51087.72
...

Users (Users.json)

users.json

```
[
  {
    "username": "UnderSinuxrein",
    "password": "4l8nYGTKMW",
    "profilePicture": "InvalidPicturePath"
  },
  {
    "username": "BlaAntigadsa",
    "password": ":Q5wjT4[e",
    "profilePicture": null
  },
  {
    "username": null,
    "password": "El[MwhxY)J",
    "profilePicture":
"src/folders/resources/images/profile/blocked/jpg/pgfMG75k4e.jpg"
  },
  {
    "username": "ScoreImmagidefon",
    "password": "Pa",
    "profilePicture":
"src/folders/resources/images/uploads/browsed/vga/n5xg70JZDw.vga"
  },
  {
    "username": "!",
    "password": "wJyfcwg*",
    "profilePicture":
"src/folders/resources/images/story/reformatted/img/hRI3TW31rC.img"
  },
  {
    "username": "WhoDenoteBel",
    "password": "ajmISQi*",
    "profilePicture":
"src/folders/resources/images/story/reformatted/jpg/y3tGM8E88H.jpg"
  },
  ...
]
```



```
Invalid User
Invalid User
Invalid User
Invalid User
Invalid User
Successfully imported User: WhoDenoteBel
...
```

XML Import

Your new colleagues have prepared some XML data for you to import.

Posts (Posts.xml)

posts.xml

```
<?xml version="1.0" encoding="utf-8"?>
<posts>
  <post>
    <caption>#everything #ring #faith #insta #infinity #swag #sunglasses
#smiley #justdoit #the #sleepless #ocean</caption>
    <user>
      <username>ScoreAntigarein</username>
    </user>
    <picture>
<path>src/folders/resources/images/story/blocked/png/1S2e13wJ3v.png</path>
    </picture>
  </post>
  <post>
    <caption>Inv</caption>
    <user>
      <username>InvalidUser</username>
    </user>
    <picture>
<path>src/folders/resources/images/story/reformatted/img/hRI3TW31rC.img</path>
    </picture>
  </post>
  <post>
    <caption>#ring #infinity #justdoit #hope #dawn</caption>
    <user>
      <username>AryaDenotehow</username>
    </user>
    <picture>
      <path>src/invalidPictures/invaliddd.img</path>
    </picture>
  </post>
  <post>
    <caption>#cool #justdoit #sky #ocean #reason #gram #faith #hope
#insta</caption>
```

```

        <user>
          <username>InvalidUser</username>
        </user>
      <picture>

<path>src/folders/resources/images/profile/blocked/jpg/pgfMG75k4e.jpg</path>
    </picture>
  </post>
  <post>
    <caption>#beautiful #swag #hope #sleepless #dusk #infinity #faith
#women #gram #ocean #epic</caption>
    <user>
      <username>WhoDenoteBel</username>
    </user>
    <picture>

<path>src/folders/resources/images/uploads/formed/jpeg/5E5ak3CkhU.jpeg</path>
    </picture>
  </post>
  <post>
    <caption>#cool #justdoit #sky #ocean #reason #gram #faith #hope
#insta</caption>
    <user>
      <username>HighAsmahow</username>
    </user>
    <picture>

<path>src/folders/resources/images/profile/blocked/jpg/pgfMG75k4e.jpg</path>
    </picture>
  </post>
  . . .
</posts>

```

```

Successfully imported Post, made by ScoreAntigarein
Invalid Post
Invalid Post
Invalid Post
Invalid Post
Successfully imported Post, made by HighAsmahow
...

```

Important: The "profilePicture", in the users.json file, and the "picture", in the posts.xml file, are both from the "Picture" entity

5. Data Export

Get ready to export the data you've imported in the previous task. Here you will have some pretty complex database querying. Export the data in the formats specified below.

Export Users with Their Posts

- Export **all users** with their **posts ordered by count of posts descending, then by user id**
- **Order the posts**, inside each user, by the **post's picture size in ascending order**
- **Format the picture size value to the 2nd digit after the floating point**
- Return the information in this format:

```
"User: {username}
Post count: {count of posts}
==Post Details:
----Caption: {caption}
----Picture Size: {size}
..."
```

```
User: ScoreAntigarein
Post count: 6
==Post Details:
----Caption: #everything #ring #faith #insta #infinity #swag #sunglasses
#smiley #justdoit #the #sleepless #ocean
----Picture Size: 10960.40
==Post Details:
----Caption: #insta #reason #sunglasses #swag #justdoit #what #dusk #morning
#madness
----Picture Size: 12210.73
...
```

Export Pictures with Size Bigger Than 30000

- Export **all pictures** with size **bigger than 30000**
- Order the result by **size ascending**
- **Format the picture size to the 2nd digit after the floating point**
- Return the information in this format:

```
"{picSize} - {picPath}
. . ."
```

```
31013.45 - src/folders/resources/images/post/blocked/img/DTy1261X2w.img
```


32495.57 - src/folders/resources/images/profile/blocked/bmp/kj0JjKpKh4.bmp
33587.94 - src/folders/resources/images/post/formed/jpg/5eEjkhYd3.jpg
33591.00 - src/folders/resources/images/mobile/infigured/bmp/03z18luJc00.bmp
33591.08 - src/folders/resources/images/mobile/infigured/bmp/03z18luJc0.bmp
33820.37 - src/folders/resources/images/mobile/blocked/digi/3ISl01969f.digi
34568.62 - src/folders/resources/images/story/infigured/png/9H9M5a59Li.png
35037.91 - src/folders/resources/images/post/formed/digi/6Ylvj97k03.digi
35756.39 - src/folders/resources/images/story/browsed/png/ubMYHgzhYb.png
39184.46 - src/folders/resources/images/uploads/timeline/jpeg/2BZvc3RXA2.jpeg
...