Peer Review Points Sale started! You will get back your PRP sp... Sale status: 40%

← Project review - SQL1 Bootcamp. Day02

Type of project	Individual
Duration	30 min
Passed Peer Reviews	0/3

Git project

ssh://git@repos-ssh.21-school.ru:2289/students/SQL_beginner._Day02.ID_574089/msa...

Open

Student

msalena@student.21-school.ru level 9

About

Introduction

The methodology of School 21 makes sense only if peer-to-peer reviews are done seriously. Please read all guidelines carefully before starting the review.

- Please, stay courteous, polite, respectful and constructive in all communications during this review.
- Highlight possible malfunctions of the work done by the person and take the time to discuss and debate it.
- Keep in mind that sometimes there can be differences in interpretation of the tasks and the scope of features. Please, stay open-minded to the vision of the other.
- If you have not finished the project yet, it is compulsory to read the entire instruction before starting the review.

Guidelines

- Evaluate only the files that are in src folder on the GIT repository of the student or group.
- Ensure to start reviewing a group project only when the team is present in full.
- Use special flags in the checklist to report, for example, an “empty work” if repository does not contain the work of the student (or group) in the src folder of the develop branch, or “cheat” in case of cheating or if the student (or group) are unable to explain their work at any time during review as well as if one of the points below is not met. However, except for cheating cases, you are encouraged to continue reviewing the project to identify the problems that caused the situation in order to avoid them at the next review.
- Doublecheck that the GIT repository is the one corresponding to the student or the group.
- Meticulously check that nothing malicious has been used to mislead you.
- In controversial cases, remember that the checklist determines only the general order of the check. The final decision on project evaluation remains with the reviewer.

Main part

Exercise 00

Checks for the file day02_ex00.sql
- The SQL script looks like below.

```
select p.name as pizzeria_name, rating
from pizzeria p left join person_visits pv on p.id = pv.pizzeria_id
where pv.id is null
```

- The result is below (raw ordering should be the same like on a screen below)

```
"DoDo Pizza"    "3.2"
```

No ✓ Yes

Exercise 01

Checks for the file day02_ex01.sql
- The SQL script looks like below.

```
select g::date as missing_date
from (select * from person_visits po where person_id in (1,2)) as po right  join
generate_series('2022-01-01','2022-01-10', interval '1 day') as g on po.visit_date =g
where po.id is null
order by 1
```

- The result is below (raw ordering should be the same like below)

```
"2022-01-03"
"2022-01-04"
"2022-01-05"
"2022-01-06"
"2022-01-07"
"2022-01-08"
"2022-01-09"
"2022-01-10"
```

No ✓ Yes

Exercise 02

Checks for the file day02_ex02.sql
- The SQL script looks like below.

```
select coalesce(p.name,'-') as person_name,
   pv.visit_date,
   coalesce(pz.name, '-') as pizzeria_name
from person p
   full join (select * from person_visits po where visit_date between '2022-01-01' and '2
022-01-03') pv on p.id = pv.person_id
   full join pizzeria pz on pz.id = pv.pizzeria_id
order by  1,2, 3
```

The result is below (raw ordering should be the same like below)

```
-       null  DinoPizza
-       null  DoDo Pizza
Andrey       2022-01-01     Dominos
Andrey       2022-01-02     Pizza Hut
Anna  2022-01-01     Pizza Hut
Denis null  -
Dmitriy      null  -
Elvira null  -
Irina  2022-01-01     Papa Johns
Kate  2022-01-03     Best Pizza
Nataly       null  -
Peter 2022-01-03     Pizza Hut
```

No ✓ Yes

Exercise 03

Checks for the file day02_ex03.sql
- The SQL script looks like below.

```
with g as (
select g::date
from generate_series('2022-01-01','2022-01-10', interval '1 day') as g)
select g::date as missing_date
from (select * from person_visits po where person_id in (1,2)) as po right  join
g as g on po.visit_date =g
where po.id is null
order by 1
```

- The result is below (raw ordering should be the same like below)

```
"2022-01-03"
"2022-01-04"
"2022-01-05"
"2022-01-06"
"2022-01-07"
"2022-01-08"
"2022-01-09"
"2022-01-10"
```

No ✓ Yes

Exercise 04

Checks for the file day02_ex04.sql
- The SQL script looks like below.

```
select pizza_name, name as pizzeria_name, price
from menu inner join pizzeria p on menu.pizzeria_id = p.id
where pizza_name in ('mushroom pizza', 'pepperoni pizza')
order by 1,2
```

- The result is below (raw ordering should be the same like below)

```
"mushroom pizza"  "Dominos"      "1100"
"mushroom pizza"  "Papa Johns"   "950"
"pepperoni pizza"  "Best Pizza"   "800"
"pepperoni pizza"  "DinoPizza"    "800"
"pepperoni pizza"  "Papa Johns"   "1000"
"pepperoni pizza"  "Pizza Hut"    "1200"
```

No [✓ Yes]

Exercise 05

Checks for the file day02_ex05.sql
- The SQL script looks like below.

```
select name
from person
where gender = 'female' and age> 25
order by 1
```

- The result is below (raw ordering should be the same like below)

```
"Elvira"
"Kate"
"Nataly"
```

No [✓ Yes]

Exercise 06

Checks for the file day02_ex06.sql
- The SQL script looks like below.

```
select m.pizza_name,  p2.name as pizzeria_name
from person_order inner join person p on p.id = person_order.person_id
inner join menu m on m.id = person_order.menu_id
inner join pizzeria p2 on m.pizzeria_id = p2.id
where p.name in ('Denis', 'Anna')
order by 1,2
```

- The result is below (raw ordering should be the same like below)

```
"cheese pizza"    "Best Pizza"
"cheese pizza"    "Pizza Hut"
"pepperoni pizza"      "Best Pizza"
"pepperoni pizza"      "DinoPizza"
"pepperoni pizza"      "Pizza Hut"
"sausage pizza"  "DinoPizza"
"supreme pizza" "Best Pizza"
```

No [✓ Yes]

Exercise 07

Checks for the file day02_ex07.sql
- The SQL script looks like below.

```
 select p.name as pizzeria_name
 from menu inner join pizzeria p on p.id = menu.pizzeria_id
     inner join person_visits pv on menu.pizzeria_id = pv.pizzeria_id
     inner join person p2 on p2.id = pv.person_id
 where price  < 800 and p2.name = 'Dmitriy' and visit_date = '2022-01-08'
```

- The result is below (raw ordering should be the same like below)

```
"Papa Johns"
```

No ✓ Yes

Exercise 08

Checks for the file day02_ex08.sql
- The SQL script looks like below.

```
select person.name
from person inner join person_order po on person.id = po.person_id
inner join menu m on m.id = po.menu_id
where gender= 'male' and pizza_name in ('pepperoni pizza','mushroom pizza')
    and  address  in ('Moscow', 'Samara')
order by 1 desc
```

- The result is below (raw ordering should be the same like below)

```
"Dmitriy"
"Andrey"
```

No ✓ Yes

Exercise 09

Checks for the file day02_ex09.sql
- The SQL script looks like below.

```
select p.name
from person p inner join person_order po on p.id = po.person_id
inner join menu m on m.id = po.menu_id
where m.pizza_name = 'pepperoni pizza' and p.gender = 'female'
intersect
select p.name
from person p inner join person_order po on p.id = po.person_id
inner join menu m on m.id = po.menu_id
where m.pizza_name = 'cheese pizza' and p.gender = 'female'
order by 1
```

- The result is below (raw ordering should be the same like below)

```
"Anna"
"Nataly"
```

No ✓ Yes

Exercise 10

Checks for the file day02_ex10.sql
- The SQL script looks like below.

```
select p1.name, p2.name, p1.address as common_address
from person p1 inner join person p2 on p1.id > p2.id and p1.address = p2.address
order by 1,2,3
```

- The result is below (raw ordering should be the same like below)

```
"Andrey" "Anna"   "Moscow"
"Denis"  "Kate"   "Kazan"
"Elvira" "Denis"  "Kazan"
"Elvira" "Kate"   "Kazan"
"Peter"  "Irina"  "Saint-Petersburg"
```

No ✓ Yes

Feedback

Fails ⓘ

Empty work Forbidden functions Cheat

Comment

Leave a comment...

✓ Review