|  |  |
| --- | --- |
| **სასწავლო კურსის სახელწოდება:** | **მონაცემთა ბაზების საწყისები (oracle)** |
| **ლექტორი:** | **ალექსანდრე ჩახვაძე** |
| **სტუდენტის სახელი და გვარი** | **დათა ჭანუყვაძე** |
| **ჯგუფის #** |  |

***შუალედური გამოცდა***

**შეფასება: 10 დავალება. თითო 3 ქულა. სულ 30 ქულა.**

**შუალედური ეყრდნობა ვირტუალურ ბოქსზე არსებულ ბაზის სქემებს.**

**მუშაობთ ამ ფაილში, სათანადო ადგილებში უთითებთ საჭირო ინფორმაციებს, სკრიპტებსა და სქრინებს. მუშაობის დასრულების შემდგომ ფაილი უნდა დააპედეეფოთ და დაარქვათ თქვენი სახელი და გვარი. მიღებული პდფ ფაილი ატვირთოთ შესაბამის ველში.**

1. dual ცხრილის დახმარებით ეკრანზე გამოიტანე დამუშავებული შენი პერსონალური ინფორმაცია. კერძოდ კი პირველ სვეტად შენი სახელი და გვარი პატარა ლათინური შიფტით სეპარირებული სფეისით, მეორე სვეტად პირველი სვეტის მნიშვნელობა დამუშავებული substr ფუქნციით პირველი 6 სიმბოლო, მესამე სვეტად პირველის სვეტის სიგრძე, მეოთხე სვეტად პირველი სვეტის სიგრძის ნაშთი 8-ზე, მეხუთე სვეტად მეოთხე სვეტის მნიშვნელობისა და პირველი სვეტის სიგრძის ჯამის ნაშთი 5-ზე.

**სკრიპტი:**

**სქრინი:**

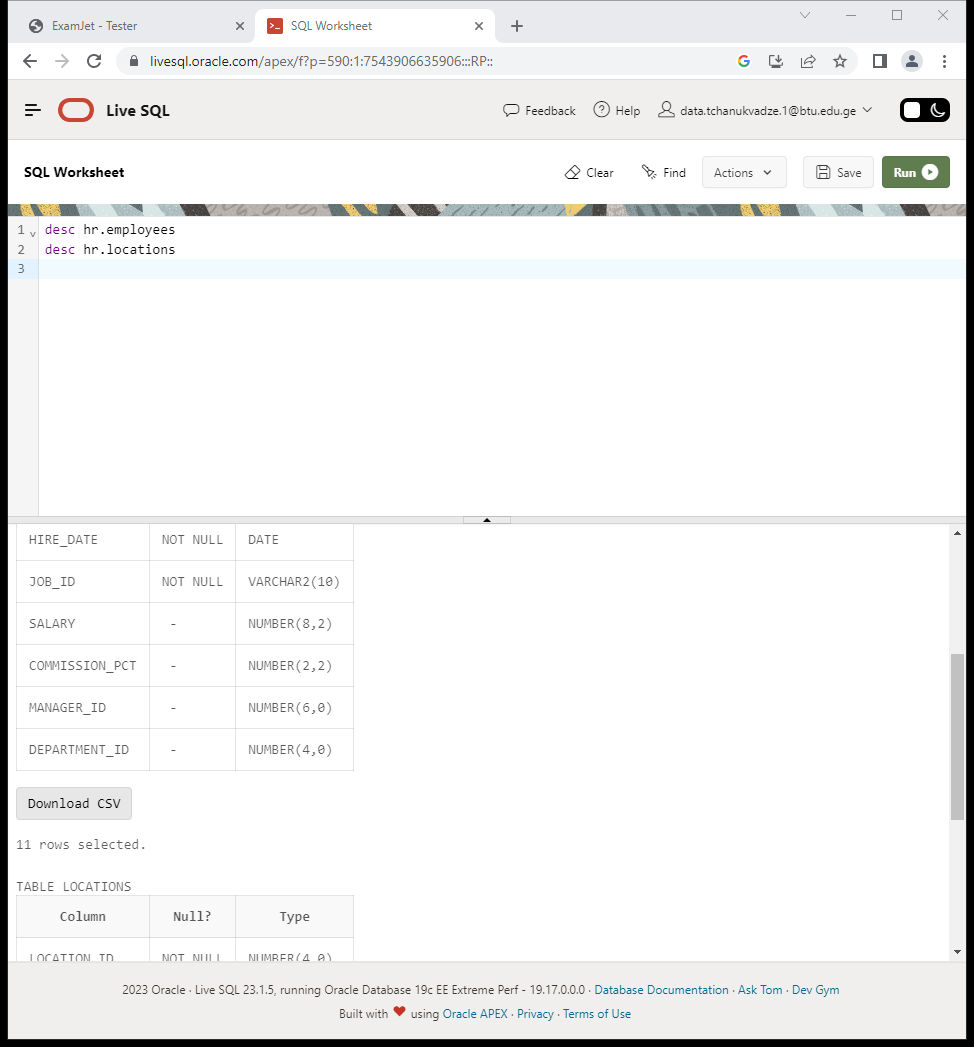
1. გამოიტანე hr სქემის employees და locations ცხრილების სტრუქტურა.

**სკრიპტი:**

desc hr.employees

desc hr.locations

**სქრინი:**

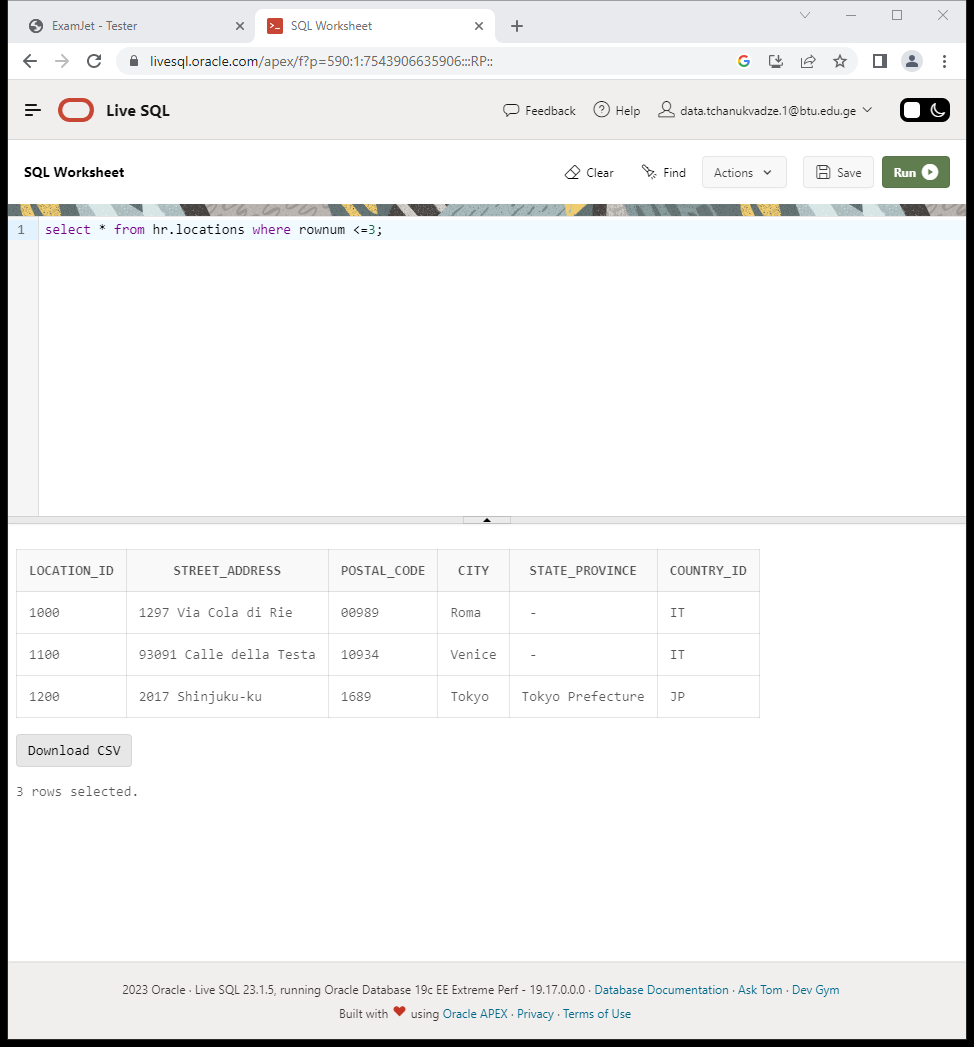
****

1. გამოიტანე hr სქემის locations ცხრილიდან ყველა სვეტი მხოლოდ პირველი 3 სტრქიონისთვის.

**სკრიპტი:**

select \* from hr.locations where rownum <=3;

**სქრინი:**

****

1. გამოიტანე hr სქემის employees ცხრილიდან თანამშროლის გვარები, ხელფასები დაფორმატებული რომ ჩანდეს ვალუტა დოლარები, სამსახურში აყვანის თარიღი გაფორმებული კერძოდ ‘კვირის დღე, რიცხვი თვე სრული სიტყვიერი წელიწადი ციფრული 4 ნიშნა’. მხოლოდ იმ სტრქიონებისთვის სადაც ხელფასი არის 3700-დან 8500-ის ჩათვლით. მონაცემები დალაგებული იყოს პირველი სვეტის კლებადობით.

**სკრიპტი:**

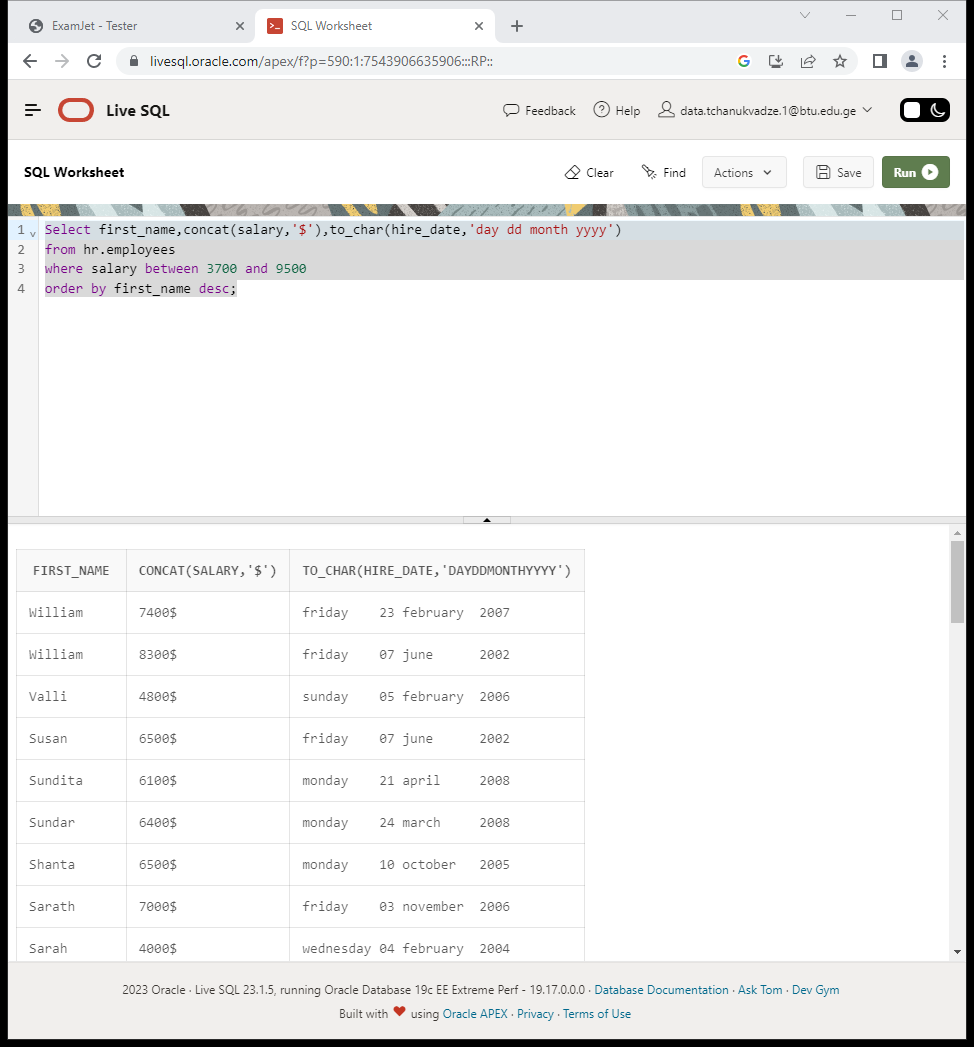
**Select first\_name,concat(salary,'$'),to\_char(hire\_date,'day dd month yyyy')**

**from hr.employees**

**where salary between 3700 and 8500**

**order by first\_name desc;**

**სქრინი:**

****

1. გამოიტანე hr სქემის employees ცხრილიდან დეპარტმენტის აიდების რაოდენობა, დეპარტმენტების აიდების საშუალო მნიშვნელობა, რომელიც იქნება დამრგვალებული მეასედებამდე. მხოლოდ იმ სტრქიონების გამოყენებით სადა გვარში მონაწილებს ან a ან m ან b ან k ან p.

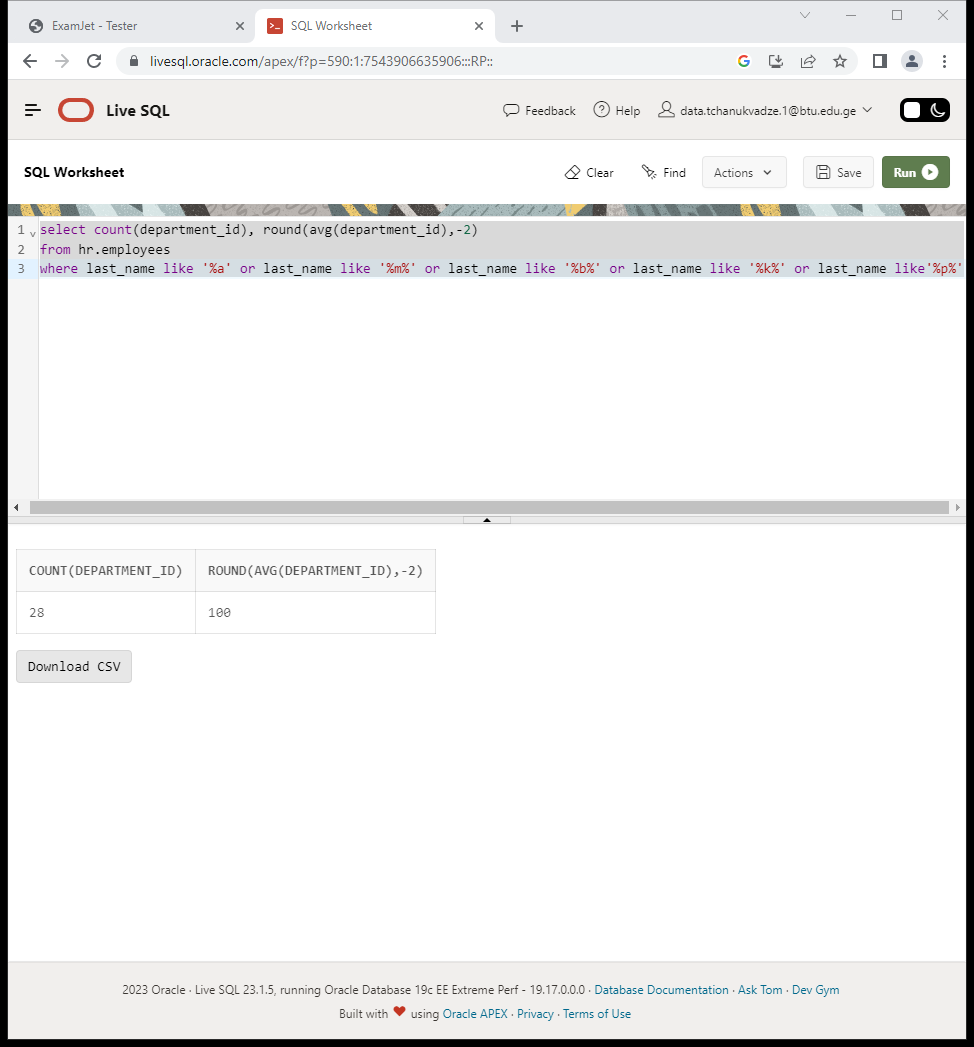
**სკრიპტი:**

**select count(department\_id), round(avg(department\_id),-2)**

**from hr.employees**

**where last\_name like '%a' or last\_name like '%m%' or last\_name like '%b%' or last\_name like '%k%' or last\_name like'%p%'**

**სქრინი:**

****

1. გამოიტანე hr სქემის employees ცხრილიდან კონკატენირებული სამსახურის აიდი და იმეილი. მხოლოდ იმ სტრიქონებისთვის სადაც გვარისა და სამსახურის აიდის სიგრძეთა ჯამი არის 5-ის ჯერადი.

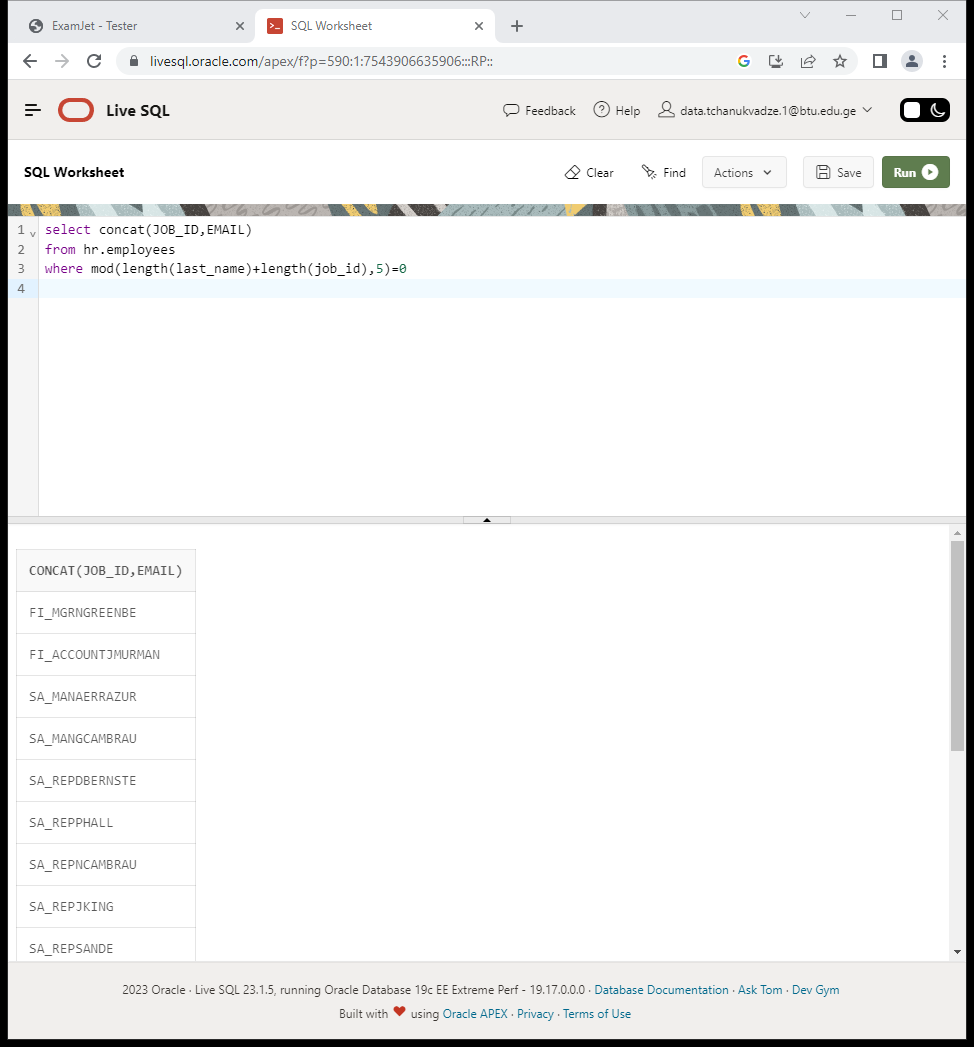
**სკრიპტი:**

**select concat(JOB\_ID,EMAIL)**

**from**

**where mod(length(last\_name)+length(job\_id),5)=0**

**სქრინი:**

****

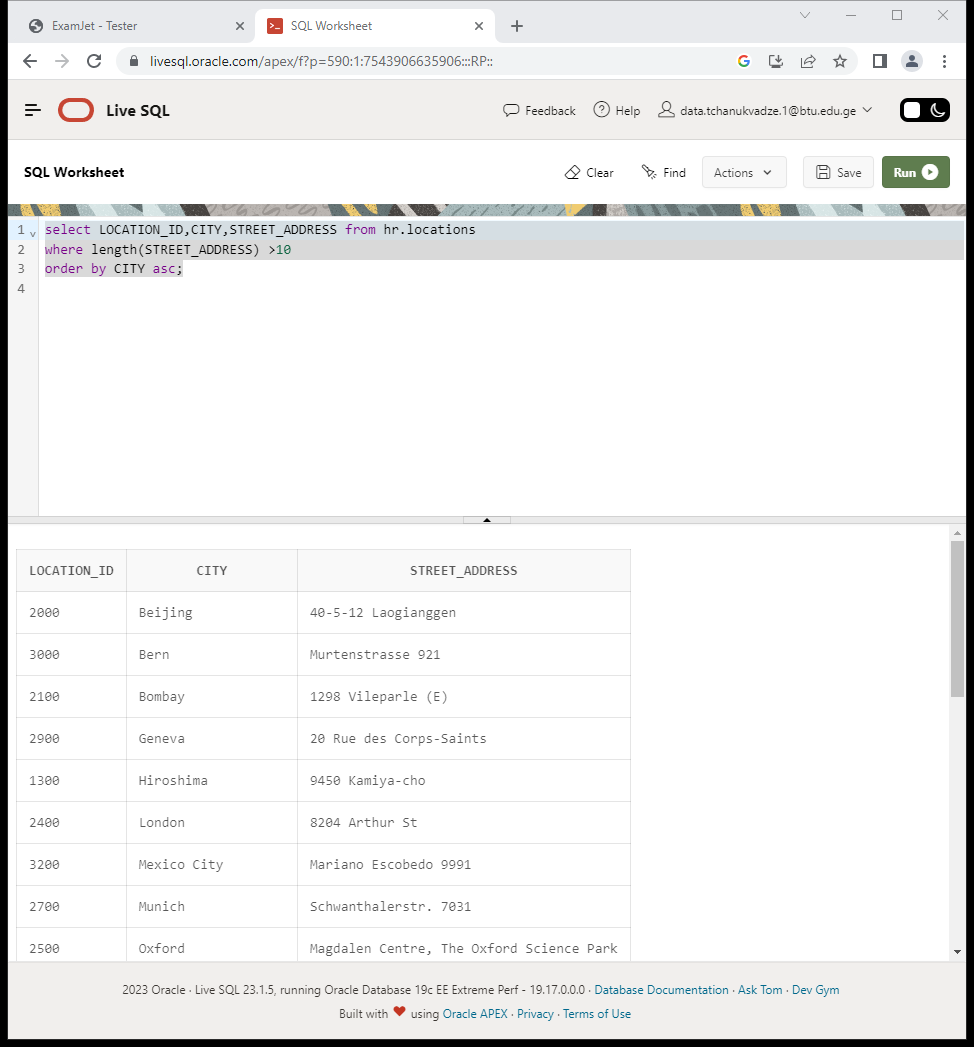
1. გამოიტანე hr სქემის locations ცხრილიდან ლოკაციის აიდი, ქალაქი, ქუჩის მისამართი, მხოლოდ იმ სტრიქონებისთვის სადაც ქუჩის მისამარტის სიგრძე არის 10 სიმბოლოზე მეტი. მონაცემები მოწესრიგდეს ქალაქის ზრდით.

**სკრიპტი:**

**select LOCATION\_ID,CITY,STREET\_ADDRESS from hr.locations**

**where length(STREET\_ADDRESS) >10**

**order by CITY asc;**

**სქრინი: **

1. გამოიტანე hr სქემის employees ცხრილიდან გვარები, ხელფასები და ხელოვნურად დაგენერირებული სვეტი სახელით uni, რომელიც გენერირდება კონკატენირებული სახელი და გვარის მე4 პოზიციიდან სტრიქონის ბოლომდე ამოღებული სიმბოლოთა ოდენობის ნაშთით 5-ზე, კერძოდ თუ ნაშთია 0 მაშინ seu, თუ ნაშთია 1 მაშინ btu, თუ ნაშთია 2 მაშინ tsu და სხვა დანარჩენ შემთხვევაში gtu. ეკრანზე ჩანდეს მხოლოდ ის სტრიქონები სადაც uni სვეტში იქნება btu და tsu. მონაცემები მოწესრიგდეს uni სვეტის ზრდადობით.

**სკრიპტი:**

**select last\_name,salary,**

**decode(mod(length(substr(concat(first\_name,last\_name),4)),5),0,'seu',1,'btu',2,'tsu','gtu') as uni**

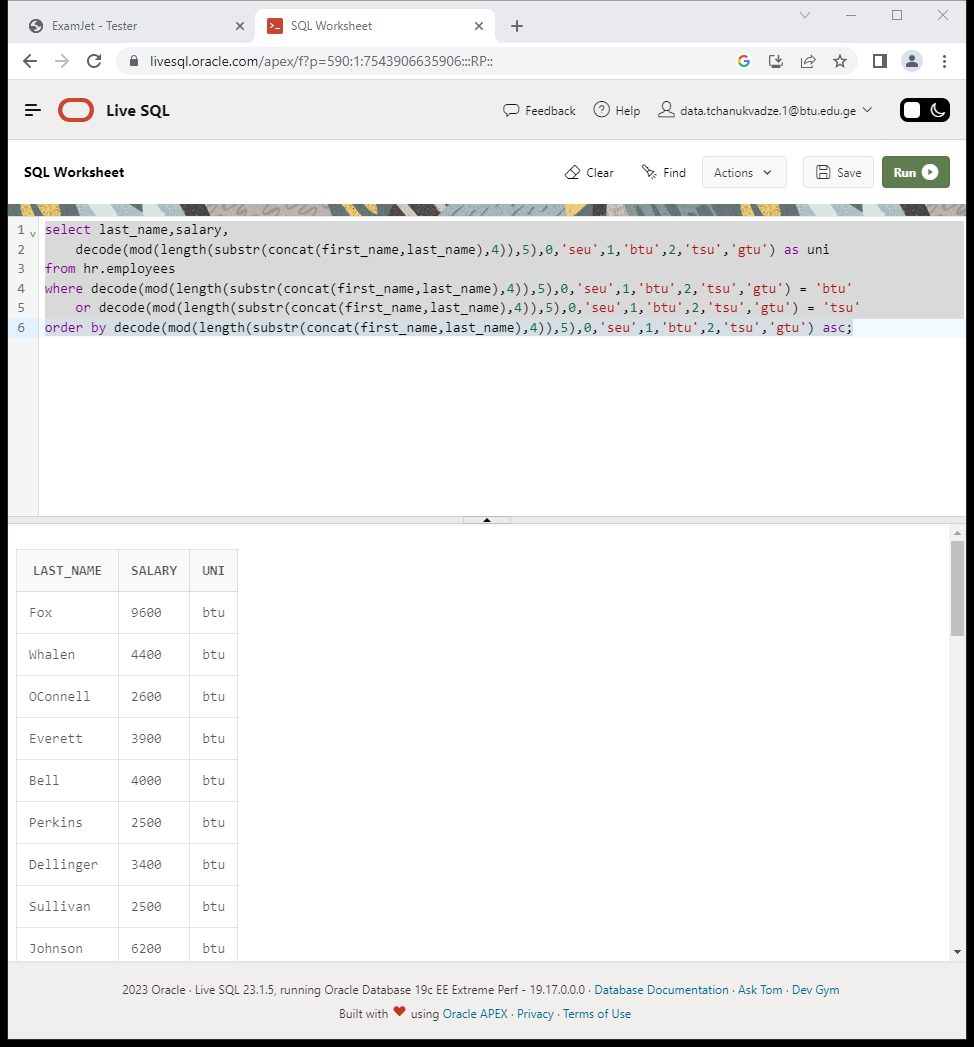
**from hr.employees**

**where decode(mod(length(substr(concat(first\_name,last\_name),4)),5),0,'seu',1,'btu',2,'tsu','gtu') = 'btu'**

**or decode(mod(length(substr(concat(first\_name,last\_name),4)),5),0,'seu',1,'btu',2,'tsu','gtu') = 'tsu'**

**order by decode(mod(length(substr(concat(first\_name,last\_name),4)),5),0,'seu',1,'btu',2,'tsu','gtu') asc;**

**სქრინი:**

****

1. გამოიტანეთ hr სქემის employees ცხრილიდან თანამშრომლის გვარის, სახელის, მეილის და ტელეფონის კონკატენირებული მნიშვნელობის სიგრძის ნაშთი 7-ზე გაყოფისას და ამ ნაშთის მქონე ჩანაწერთა ოდენობა, ამისთვის გამოიყენებ მონაცემთა დაჯგუფება. ეკრანზე არ ჩანდეს ის ჯგუფი სადაც ნაშთი მიირება 0. მონაცემები მოწესრიგდეს მიღებული ნაშთების ზრდით.

**სკრიპტი:**

**სქრინი:**

1. რეპორტი მოითხოვს გატანილ იქნეს ეკრანზე თანამშრომლის გვარი, ხელფასი და ქვეყანა, ამისთვის უნდა გამოიყენოთ hr სქემის employees და countries ცხრილები. მათ დასაკავშირებლად გამოიყენეთ join on ქლოუზი და გადაბმის ლოგიკა კი იყოს ნამაშრომელთა გვარის სიგრძე დაემთხვეს ქვეყნის სახელის სიგრძეს. მონაცემები მოწესრიგდეს თანამშრომელთა გვარის ზრდით.

**სკრიპტი:**

**select e.last\_name,e.salary,c.country\_name**

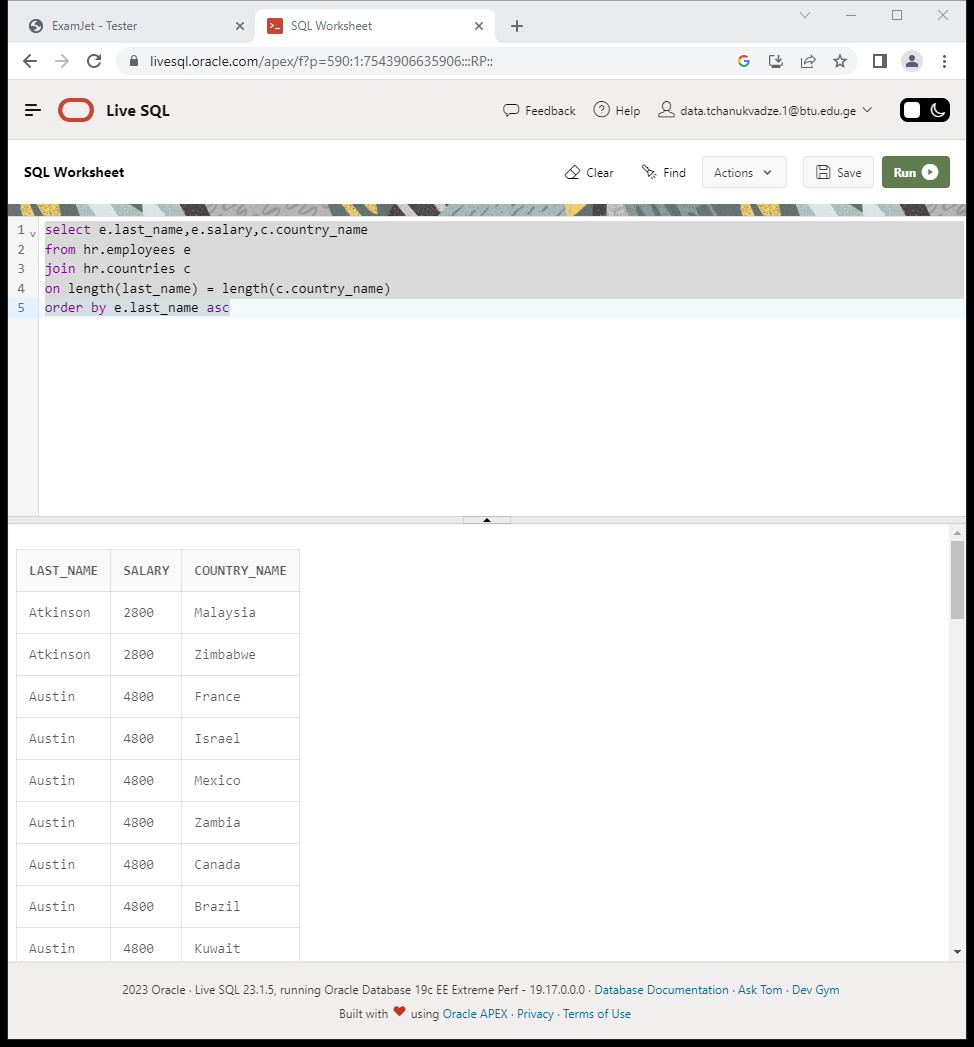
**from hr.employees e**

**join hr.countries c**

**on length(last\_name) = length(c.country\_name)**

**order by e.last\_name asc**

**სქრინი:**

****