Atskaite

1. Pointers

1) C#

Programma:

```
class Agent

{

    public string name = "";
}



static void Main(string[] args)
{
    var agent_1 = new Agent(); // Agent* agent_1 = new Agent();

    agent_1.name = "a";


    var agent_2 = new Agent(); // Agent* agent_2 = new Agent();

    agent_2.name = "b";


    agent_1 = agent_2;

    agent_2.name = "c";

    Console.WriteLine(agent_1.name + agent_2.name);
}
```

Tiks izvadīts rezultāts cc, jo dati atrodas stack atmiņā un abi norāda uz vienu un to pašu objektu heapā

Divas programmas:

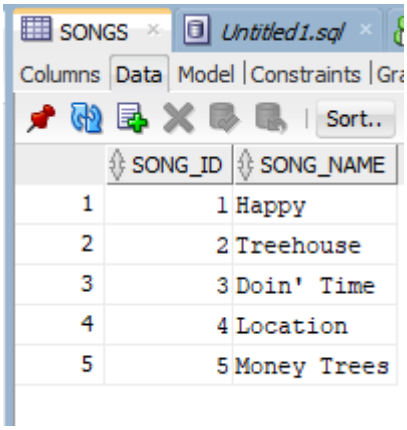
1) Šeit tiks izvadīts 3, jo tiek lietots value type

```
int x = 3;
int y = x
 y = 4;
Console.WriteLine(x);
```

2) Šeit tiks izvadīts 4, tāpēc, ka tiek lietots reference type un tiek izmantoti rādītāji, kas norāda uz vienu un to pašu objektu

```
class MyInt
  {
      public int MyValue;
  }
MyInt x = new MyInt();
x.MyValue = 3;
MyInt y = new MyInt();
y = x;
y.MyValue = 4;
Console.WriteLine (x.MyValue);
```

2. SQL

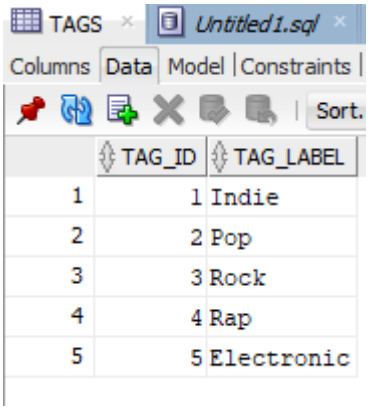


SONGS | Untitled1.sql

Columns | Data | Model | Constraints | Gra

Sort..

	SONG_ID	SONG_NAME
1	1	Happy
2	2	Treehouse
3	3	Doin' Time
4	4	Location
5	5	Money Trees

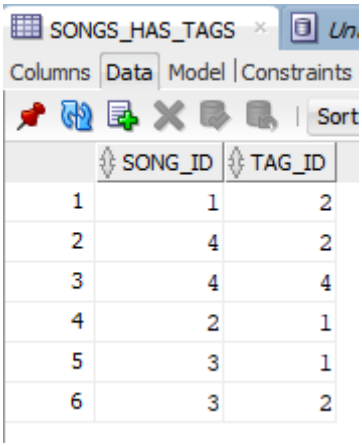


TAGS | Untitled1.sql

Columns | Data | Model | Constraints |

Sort.

	TAG_ID	TAG_LABEL
1	1	Indie
2	2	Pop
3	3	Rock
4	4	Rap
5	5	Electronic



SONGS_HAS_TAGS | Un

Columns | Data | Model | Constraints

Sort

	SONG_ID	TAG_ID
1	1	2
2	4	2
3	4	4
4	2	1
5	3	1
6	3	2

Tiek izvadītas visas dziesmas, kuras atzīmētas ar tag "Pop" žanrs

Worksheet | Query Builder

```
select c.tag_label as TAG,s.song_name as SONG
from TAGS c
  inner join SONGS_HAS_TAGS  sc on (sc.tag_id = c.tag_id)
  inner join Songs  s on (s.song_id = sc.song_id)
where
  (c.tag_label = 'Pop');
```

Query Result

SQL | All Rows Fetched: 3 in 0,018 seconds

	TAG	SONG
1	Pop	Happy
2	Pop	Doin' Time
3	Pop	Location