

# Problem 2004: The Number of Seniors and Juniors to Join the Company

## Problem Information

Difficulty: **Hard**

Acceptance Rate: 0.00%

Paid Only: No

## Problem Description

Table:

Candidates

+-----+-----+ | Column Name | Type | +-----+-----+ | employee\_id | int | |  
experience | enum | | salary | int | +-----+-----+ employee\_id is the column with unique  
values for this table. experience is an ENUM (category) type of values ('Senior', 'Junior'). Each  
row of this table indicates the id of a candidate, their monthly salary, and their experience.

A company wants to hire new employees. The budget of the company for the salaries is

\$70000

. The company's criteria for hiring are:

Hiring the largest number of seniors.

After hiring the maximum number of seniors, use the remaining budget to hire the largest number of juniors.

Write a solution to find the number of seniors and juniors hired under the mentioned criteria.

Return the result table in

any order

The result format is in the following example.

Example 1:

Input:

Candidates table: +-----+-----+-----+ | employee\_id | experience | salary |  
+-----+-----+-----+ | 1 | Junior | 10000 | | 9 | Junior | 10000 | | 2 | Senior | 20000 | |  
11 | Senior | 20000 | | 13 | Senior | 50000 | | 4 | Junior | 40000 | +-----+-----+-----+

Output:

+-----+-----+ | experience | accepted\_candidates |  
+-----+-----+ | Senior | 2 | | Junior | 2 | +-----+-----+

Explanation:

We can hire 2 seniors with IDs (2, 11). Since the budget is \$70000 and the sum of their salaries is \$40000, we still have \$30000 but they are not enough to hire the senior candidate with ID 13. We can hire 2 juniors with IDs (1, 9). Since the remaining budget is \$30000 and the sum of their salaries is \$20000, we still have \$10000 but they are not enough to hire the junior candidate with ID 4.

Example 2:

Input:

Candidates table: +-----+-----+-----+ | employee\_id | experience | salary |  
+-----+-----+-----+ | 1 | Junior | 10000 | | 9 | Junior | 10000 | | 2 | Senior | 80000 | |  
11 | Senior | 80000 | | 13 | Senior | 80000 | | 4 | Junior | 40000 | +-----+-----+-----+

Output:

+-----+-----+ | experience | accepted\_candidates |  
+-----+-----+ | Senior | 0 | | Junior | 3 | +-----+-----+

Explanation:

We cannot hire any seniors with the current budget as we need at least \$80000 to hire one senior. We can hire all three juniors with the remaining budget.

## Code Snippets

### MySQL:

```
# Write your MySQL query statement below
```

### MS SQL Server:

```
/* Write your T-SQL query statement below */
```

### PostgreSQL:

```
-- Write your PostgreSQL query statement below
```

### Oracle:

```
/* Write your PL/SQL query statement below */
```

### Pandas:

```
import pandas as pd

def count_seniors_and_juniors(candidates: pd.DataFrame) -> pd.DataFrame:
```

## Solutions

### MySQL Solution:

```
# Write your MySQL query statement below
```

### MS SQL Server Solution:

```
/* Write your T-SQL query statement below */
```

### PostgreSQL Solution:

```
-- Write your PostgreSQL query statement below
```

### Oracle Solution:

```
/* Write your PL/SQL query statement below */
```

### Pandas Solution:

```
import pandas as pd

def count_seniors_and_juniors(candidates: pd.DataFrame) -> pd.DataFrame:
```