

# BRIQUE 코딩테스트

## 주의사항

- 문제는 총 8개로, 문제 1번 ~ 5번은 필수로 풀어 주시고, 문제 6번~8번은 3문제 중 1개 문제를 선택하여 풀어 주시면 됩니다.
- 해결한 문제의 결과는 실행되는 프로그램의 형태여야 합니다. 즉, 일부 기능만 포함하는 소스코드의 일부 및 함수 등의 형태로 제출하는 것은 곤란합니다.
- 해결한 문제는 소스코드를 포함한 프로젝트 파일을 함께 제출하셔야 합니다. 즉, 평가자가 개발도구를 이용하여 직접 소스코드를 컴파일하고 그 결과물을 수행해 볼 수 있는 상태로 제출해 주세요. 필요 시 컴파일 및 실행 환경에 대한 설명을 첨부해 주시기 바랍니다.
- 외부 라이브러리를 사용한 경우, 사용된 라이브러리도 함께 제출해 주세요.
- “가산점” 항목이 포함된 문제의 경우, “가산점” 항목을 포함하여 문제를 해결하면 가산점이 주어집니다. 반드시 “가산점” 항목을 포함하여 문제를 해결해야 하는 것은 아니니, 본인의 판단 하에 풀이 방법을 선택하세요.
- 문제 풀이에 사용하는 언어 및 개발환경에 대해서는 기본적으로 제약이 없으나, 가능하면 본인이 지원하는 분야에 해당하는 언어/개발환경을 사용하시면 평가에 긍정적인 감안요소로 작용합니다. (예 : Web UI 기반의 Front-end 개발 지원자는 Javascript 활용시 우대)
- 코딩테스트는 단순히 원하는 결과가 도출되었는가를 보는 것이 아니라, 문제 해결을 위한 접근방법, 코드 구성, 결과물 표현 등 문제 풀이의 전 과정을 통해 주어진 업무를 얼마나 잘 수행할 수 있는가를 종합적으로 평가하는 데 사용됩니다. 응시자는 이러한 평가 기준을 숙지하시고 테스트에 응해 주시기 바랍니다.

## [문제 1]

아래 csv 파일을 읽어서 필요한 결과를 출력하시오.

[https://drive.google.com/file/d/1acBm6xVe9g2gyDBXR2ub9NuT\\_tpW5EwI/view?usp=sharing](https://drive.google.com/file/d/1acBm6xVe9g2gyDBXR2ub9NuT_tpW5EwI/view?usp=sharing)

csv 파일을 읽어서 오류가 있는 항목들을 걸러내고, 필요한 내용이 있는 항목에 대해서 계산 결과를 하여 출력한다.

- 제약조건

- 각 라인별로 계산해야 하는 항목

- 최소값
- 최대값
- 합계
- 평균
- 표준편차
- 중간값

- 숫자에 해당하는 값만 계산 결과를 출력하고 숫자가 아닌 값이 하나라도 존재하면 해당 행에 대해서는 계산 결과를 출력하지 않는다.

- 출력 후 숫자가 아닌 값들만 모아서 출력해본다.

### 실행 예

```
...
3.0 84.0 334.0 33.4 28.82976856576472 22.5
9.0 98.0 579.0 57.9 33.785105199382365 68.5
24.0 99.0 645.0 64.5 30.580494872821575 70.0
10.0 88.0 536.0 53.6 23.24841691145633 53.5
5.0 93.0 637.0 63.7 25.077878698167435 70.5
3.0 98.0 533.0 53.3 28.035889697156232 58.0
4.0 97.0 526.0 52.6 35.81185402752682 49.5
14.0 85.0 363.0 36.3 23.6363655788655 32.0
11.0 90.0 570.0 57.0 31.481917208313586 62.5
7.0 92.0 368.0 36.8 23.313324754549942 31.5
4.0 94.0 323.0 32.3 25.302832516011588 29.5
```

-----  
The total number of lines:

The calculated lines:

The error values:

## [문제 2]

클라이언트 프로그램이 서버 프로그램에 Ping 을 보내면 Pong 을 응답받는 프로그램 작성하시오.

### - 제약조건

- TCP, UDP 중 택일
- 클라이언트에서 보낸 메시지가 Ping 일 경우에 Pong 응답을 한다.
- 클라이언트에서 보낸 메시지가 Ping 이 아니면 클라이언트에서 보낸 메시지를 그대로 응답을 한다.

### - 가산점

- 비동기 요청/응답을 수행하도록 프로그램을 작성.
  - client는 요청을 보낸 후 server의 응답을 기다리지 않고 바로 다음 요청을 보낼 수 있다.
  - server는 client로부터 요청을 받은 후, 3초간 기다렸다가 응답을 보낸다.

#### 서버 실행 예 (동기식 요청/응답)

Connected by ('127.0.0.1', 56768)  
Received: Ping  
Send: Pong  
Received: Ping  
Send: Pong  
Received: foobar  
Send: foobar

#### 클라이언트 실행 예

Send: Ping  
Received: Pong  
Send: Ping  
Received: Pong  
Send: foobar  
Received: foobar

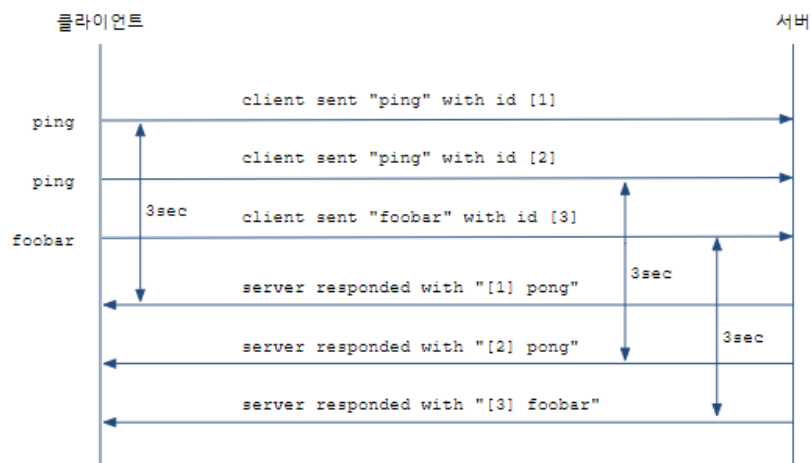
#### 서버 실행 예 (비동기식 요청/응답)

Connected by ('127.0.0.1', 56768)  
Received(1): Ping  
Received(2): Ping  
Received(3): foobar  
Send: Pong (1)  
Send: Pong (2)  
Send: foobar (3)

#### 클라이언트 실행 예

Send(1): Ping  
Send(2): Ping  
Send(3): foobar  
Received: Pong (1)  
Received: Pong (2)  
Received: foobar (3)

### 비동기식 요청/응답 Flow



### [문제 3]

주어진 접속정보를 이용하여 DB에 접속해보면 아래 ER diagram의 형태로 된 Table들이 있다. 이 테이블들을 분석하여 다음 조건에 맞는 결과를 출력해본다.

#### - 제약조건

- 출력 열은 총 9개이다.

- 종업원 번호 (emp\_no)
- 이름 (first\_name)
- 성 (last\_name)
- 성별 (gender)
- 고용일자 (hire\_date)
- 부서 이름 (dept\_name)
- 직급 (title)
- 최대 급여 (max\_salary)

- 2000년 이후 고용된 종업원들을 대상으로 한다

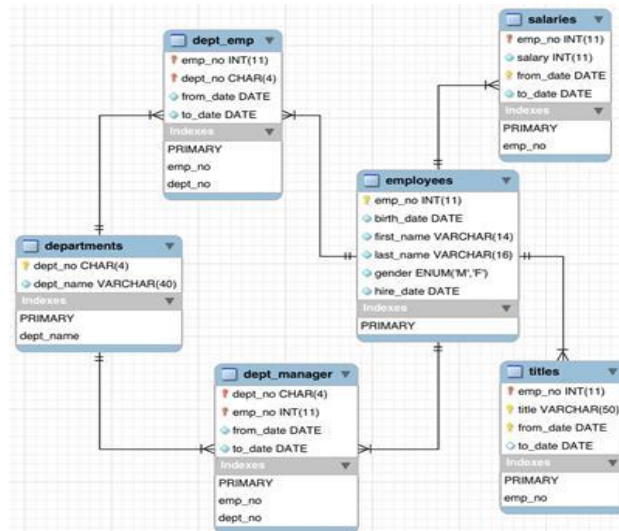
- 최대 급여는 각 종업원들이 받았던 급여 중 가장 높은 값을 의미한다.

- 문제풀이의 결과는 SQL query 만이 아닌 프로그램이다.

#### 접속정보

host: db.brique.kr  
username: codingtest  
password: 12brique!@  
db name: employees  
db type : MariaDB

#### ER Diagram



#### 실행 예

emp_no	first_name	last_name	gender	hire_date	dept_name	title
463807	Bikash	Covnot	M	2000-01-28	Quality Management	Engineer
205048	Ennio	Alblas	F	2000-01-06	Finance	Senior Staff
499553	Hideyuki	Delgrande	F	2000-01-22	Development	Engineer
422990	Jaana	Verspoor	F	2000-01-11	Development	Engineer
424445	Jeong	Boreale	M	2000-01-03	Development	Engineer
108201	Mariangiola	Boreale	M	2000-01-01	Production	Senior Engineer
72329	Randi	Luit	F	2000-01-02	Human Resources	Staff
60134	Seshu	Rathonyi	F	2000-01-02	Customer Service	Staff
60134	Seshu	Rathonyi	F	2000-01-02	Marketing	Staff
227544	Shahab	Demeyer	M	2000-01-08	Human Resources	Senior Staff
47291	Ulf	Flexer	M	2000-01-12	Human Resources	Staff
222965	Volkmar	Perko	F	2000-01-13	Human Resources	Senior Staff
226633	Xuejun	Benzmuller	F	2000-01-04	Sales	Staff
226633	Xuejun	Benzmuller	F	2000-01-04	Marketing	Staff
428377	Yucai	Gerlach	M	2000-01-23	Production	Engineer

15 rows in set

#### [문제 4]

아래의 표와 같이 기온과 습도를 입력 받을 수 있도록 입력 테이블을 만들고 그 입력 값에 대해 아래 그래프와 같은 형태의 선격은선 그래프를 그리시오.

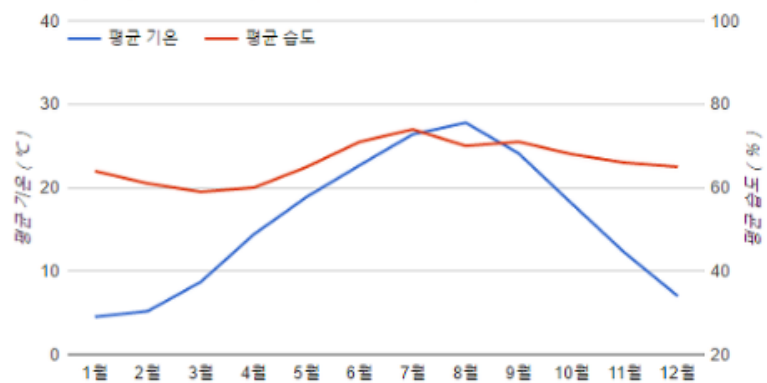
- 제약조건
  - Chart libraries를 사용할 수 있다.
  - 입력은 계속 변경 될 수 있으며 그래프에 바로 반영 되어야 한다.

#### - 가산점

- 입력 테이블 아래에 " random " 버튼을 만들고, " random " 버튼 클릭 시 자동으로 입력 테이블의 " 기온 ", " 습도 " 항목에 임의의 값이 입력된다. 입력이 변경되면 자동으로 chart에 변경된 값이 반영 된다.

#### 실행 예

	평균 기온	평균 습도
1월	4.5	64
2월	5.2	61
3월	8.7	59
4월	14.4	60
5월	18.9	65
6월	22.7	71
7월	26.4	74
8월	27.8	70
9월	24.1	71
10월	18.1	68
11월	12.2	66
12월	7	65



## [문제 5]

호출할 때마다 랜덤한 response 를 주는 서버가 있다. 이 서버를 100번 호출하여 각각의 결과 값들이 몇 번 반복 되었는지 계산한다.

- 제약조건
  - 마지막에 각 횟수들의 총 합을 출력한다.
  - 가장 많은 횟수가 나온 값부터 출력한다.

### 접속할 사이트 주소

- <https://gturnquist-quoters.cfapps.io/api/random>

### 실행 예

```
count: 15 {'id': 1, 'quote': 'Working with Spring Boot is like pair-programming with the Spring developers.'}
count: 12 {'id': 9, 'quote': 'So easy it is to switch container in #springboot.'}
count: 12 {'id': 6, 'quote': 'It embraces convention over configuration ... such as Ruby on Rails.'}
count: 9 {'id': 8, 'quote': 'I don't worry about my code scaling ... keeping the conventions that just work.'}
count: 8 {'id': 11, 'quote': 'I have two hours today to build an app from scratch. @springboot to the rescue!'}
count: 8 {'id': 7, 'quote': 'The real benefit of Boot, however ..., I know it's a safe bet.'}
count: 8 {'id': 4, 'quote': 'Previous to Spring Boot ... and many hours of frustration.'}
count: 7 {'id': 10, 'quote': 'Really loving Spring Boot, makes stand alone Spring apps easy.'}
count: 7 {'id': 5, 'quote': 'Spring Boot solves this problem ... just to figure out how it's all pieced together.'}
count: 6 {'id': 12, 'quote': '@springboot with @springframework is pure productivity! ... #newFavLib'}
count: 5 {'id': 3, 'quote': 'Spring has come quite a ways in addressing ... I built an application using it.'}
count: 3 {'id': 2, 'quote': 'With Boot you deploy everywhere you can find a JVM basically.'}
```

Total count: 100

### [문제 6]

수평 직선의 지표면에 탑  $N$ 개를 세웠다. 모든 탑의 꼭대기에는 지표면과 수평하게 직선으로 발사되는 레이저 신호를 송신하는 장치가 설치 되어 있고 발사한 신호는 신호를 보낸 탑보다 높은 탑의 몸체에서만 수신한다. 또한 한 번 수신된 신호는 다른 탑으로 송신되지 않는다.

예를 들어 높이가 6,9,5,7,4인 다섯 탑이 자기 기준으로 왼쪽으로 동시에 레이저 신호를 발사한다. 그러면, 탑은 다음과 같이 신호를 주고받는다. 높이가 4인 다섯번째 탑에서 발사한 신호는 높이가 7인 네 번째 탑이 수신하고, 높이가 7인 네번째 탑의 신호는 높이가 9인 두 번째 탑이, 높이가 5인 세 번째 탑의 신호도 높이가 9인 두 번째 탑이 수신한다. 높이가 9인 두 번째 탑과 높이가 6인 첫 번째 탑이 보낸 레이저 신호는 어떤 탑에서도 수신할 수 없다.

$N$ 개의 탑의 높이를 입력 받았을 때, 각각의 탑에서 발사한 신호를 어느 탑에서 수신하는지를 알아내는 프로그램을 작성한다.

#### 실행 예

```
Input: 6 9 5 7 4
result: 0 0 2 2 4
```

### [문제 7]

괄호가 알맞게 짝지어진 가장 긴 부분의 길이를 구하시오. ‘(‘와 ‘)’로만 이루어진 문자열에서, 괄호가 알맞게 짝지어진 가장 긴 부분의 길이를 구한다. “(())”의 경우 가장 긴 유효한 부분은 “()” 이므로 길이는 2 이다. “())()”의 경우는, 가장 긴 유효한 부분은 “()()” 이므로 길이는 4 이다.

#### 실행 예

```
Input: ) ( ) ( )
result: 4
```

```
Input: ( ( )
result: 2
```

### [문제 8]

1번부터  $N$ 번까지  $N$ 명의 사람이 원을 이루면서 앉아있고, 양의 정수  $M(\leq N)$ 이 주어진다. 이제 순서대로  $M$ 번째 사람을 제거한다. 한 사람이 제거되면 남은 사람들로 이루어진 원을 따라 이 과정을 계속해 나간다. 이 과정은  $N$ 명의 사람이 모두 제거될 때까지 계속된다. 원에서 사람들이 제거되는 순서를  $(N, M)$  - 조세퍼스 순열이라고 한다. 예를 들어  $(7, 3)$  - 조세퍼스 순열은  $\langle 3, 6, 2, 7, 5, 1, 4 \rangle$ 이다.

$N$ 과  $M$ 이 주어지면  $(N, M)$  - 조세퍼스 순열을 구하는 프로그램을 작성하시오.

#### 실행 예

```
Input: 7 3
result: <3, 6, 2, 7, 5, 1, 4>
```