

BÀI TẬP THỰC HÀNH SỐ 4

THỰC HÀNH TẠO DỮ LIỆU TEST – DUMP DATA

I. MỤC TIÊU

Kết thúc bài thực hành số 4, các nhóm phải hoàn thành các nhiệm vụ sau:

- Phân tích và thiết kế CSDL (viết báo cáo theo mẫu Lab 3 của CNPM)
- Phân tích và thiết kế Giao diện (viết báo cáo theo mẫu Lab 4 của CNPM), và design trực tiếp trên MVC
- Tạo dữ liệu test cho ứng dụng ở database (dump data)
- Tạo dữ liệu test cho ứng dụng ở javascript và trong source code

II. CƠ SỞ LÝ THUYẾT

2.1. Tạo dữ liệu test cho ứng dụng ở database

2.1.1. Mục đích tạo dữ liệu dump data

Trong quá trình xây dựng ứng dụng, chúng ta sẽ cần có tập dữ liệu sạch (số lượng tương đối lớn) để:

- **Database Test:**
 - Test các câu query database (trả lời kết quả có đúng không) so với filter dữ liệu trực tiếp trên file excel, trước khi tích hợp câu query vào code.
 - Kiểm tra mức độ hợp lý trong thiết kế của các bảng (có dễ lấy dữ liệu không, thời gian lấy dữ liệu có nhanh không, ...)
- **Functional Test:** tối thiểu 1k bản ghi
 - Test các form nhập liệu (các thao tác insert, update, delete, search, view)
 - Test các form báo cáo
 - Test các thao tác xử lý bên trong hệ thống (bao gồm cả trigger, procedure)
- **Non-functional Test:** bao gồm
 - Performance test: test hiệu suất của hệ thống
 - Stress test: test các giới hạn và mức độ ổn định của hệ thống
 - Load test: test khả năng chịu tải của hệ thống
- User Acceptance Test – UAT: test nghiệm thu ứng dụng
- Tạo ứng dụng demo

2.1.2. Quy trình tạo dữ liệu dump data

Căn cứ vào:

- Biểu mẫu / giao diện (interface)
- Cấu trúc các table trong database (các trường dữ liệu, kiểu dữ liệu, khóa chính, khóa phụ, ràng buộc giữa các bảng)

Các bước thực hiện:

1. Sinh dữ liệu cho các bảng tham số, bảng category
2. Sinh dữ liệu cho các bảng không có FK
3. Sinh dữ liệu cho các bảng có FK

Lưu ý:

- **Với các bảng không có FK:** tạo dữ liệu theo cấu trúc của bảng → download file dữ liệu → import file dữ liệu vào database
- **Với các bảng có FK:** dùng tool tạo dữ liệu (trừ trường FK) → download file dữ liệu → chèn thêm cột FK với giá trị của FK tương ứng trong bảng có PK là FK của bảng → import file dữ liệu vào database

2.1.3. Công cụ sinh dữ liệu tự động

Có rất nhiều công cụ để sinh dữ liệu tự động: công cụ online, và công cụ cài đặt. Trong bài thực hành này, chúng ta sẽ sử dụng công cụ online mockaroo (<https://www.mockaroo.com/>) để tạo dummy data cho database.

Công cụ này cho phép tạo miễn phí 1000 bản ghi dữ liệu ứng với mỗi bảng, và có thể lưu dữ liệu dưới nhiều định dạng (CSV, JSON, SQL, XML, và excel).

Thêm các field dữ liệu theo cấu trúc như đã định nghĩa trong TABLE cần sinh dữ liệu (cùng tên, kiểu dữ liệu, kích thước dữ liệu, có được NULL hay không)

Mockaroo lets you generate up to 1,000 rows of realistic test data in CSV, JSON, SQL, and Excel formats.
Need more data? Plans start at just \$50/year.

Field Name	Type	Options
id	Row Number	blank: 0 % <i>fx</i> ×
first_name	First Name	blank: 0 % <i>fx</i> ×
last_name	Last Name	blank: 0 % <i>fx</i> ×
email	Email Address	blank: 0 % <i>fx</i> ×
gender	Gender	blank: 0 % <i>fx</i> ×
ip_address	IP Address	blank: 0 % <i>fx</i> ×

Add another field

Rows: 1000 Format: **CSV** Line Ending: Unix (LF) Include: ☒ header ☐ BOM

Download Data Preview More Want to save this for later? [Sign up for free.](#)

Các kiểu dữ liệu mà ứng dụng hỗ trợ:

Choose a Type Find Type...

All (122) Basic (26) IT (20) Personal (28) Location (14) Money (8) Health (17) Advanced (8)

App Bundle ID <i>Three part app bundle id:</i> com.google.powerflex com.microsoft.prodder	App Name <i>Fake app names</i>	App Version <i>Random 2 and 3 part app version numbers</i>
Avatar <i>Random avatar image url from Robohash</i>	Base64 Image URL <i>Base64 encoded image urls like: data:image/png;base64,iVBORw...</i>	Binomial Distribution <i>Generates numbers based on a binomial distribution with a specific probability of success.</i>
Bitcoin Address 1EZ5PdVcsVEaaKYH37t8toLodJc97eooY6 1HpzKTCbcj57vkoQZCfhjyvZK97MRzktVk 1BPny1b66pdZuZC87tjikS8EG5CHRYp4ni	Blank <i>Always generates a null value</i>	Boolean true false
Buzzword contextually-based radical proactive	Catch Phrase <i>Multiple buzzwords strung together</i>	City New York Berlin London

Can't find what you're looking for? [You can mock almost anything using Regular Expressions...](#) Close

Sau khi sinh dữ liệu, chúng ta có các lựa chọn sau:

[Download Data](#)
[Preview](#)
[More ▾](#)
[Want to save this for later? Sign up for free.](#)

[Follow @mocarodev](#)

Clone this Schema
 Delete this Schema
 Import fields from Excel/CSV header...
 Import fields from CREATE TABLE sql...

Why is test data im

- **Preview:** xem file dữ liệu đã sinh ra trước khi chọn các thao tác khác.
- **Download data:** lưu file dữ liệu dưới định dạng đã chọn (CVS, Excel, XML, SQL, JSON, ...)
- **Save this for later:** Lưu vào tài khoản để sử dụng lần sau (cần tạo tài khoản và đăng nhập trước đó)
- **Clone this Schema:** Nhân bản bảng dữ liệu với cấu trúc tương tự
- **Delete this Schema:** Xóa cấu trúc bảng dữ liệu đã tạo (trong trường hợp chọn kiểu file dữ liệu là SQL với tùy chọn **Include create table**).

Rows: Format: Table Name: ☐ include create table

- **Import fields from Excel/CSV header:** tạo nhanh các field dựa trên tên các cột dữ liệu trong file Excel/CSV
- **Import fields from CREATE TABLE sql:** tạo nhanh các field dựa trên tên các field trong câu lệnh CREATE TABLE

Download file dữ liệu, và thực hiện việc import dữ liệu theo các bước như đã nêu ở 2.1.2. Quy trình tạo dữ liệu dump data

Ví dụ: tạo 50 record cho bảng Students như sau:

Field Name	Type	Options
<input type="text" value="Id"/>	Row Number	blank: <input type="text" value="0"/> % <input type="text" value="fx"/> ×
<input type="text" value="Name"/>	Full Name	blank: <input type="text" value="0"/> % <input type="text" value="fx"/> ×
<input type="text" value="Username"/>	Username	blank: <input type="text" value="0"/> % <input type="text" value="fx"/> ×
<input type="text" value="Password"/>	Password	blank: <input type="text" value="0"/> % <input type="text" value="fx"/> ×
<input type="text" value="Email"/>	Email Address	blank: <input type="text" value="0"/> % <input type="text" value="fx"/> ×
<input type="text" value="Age"/>	Number	min: <input type="text" value="1"/> max: <input type="text" value="100"/> decimals: <input type="text" value="0"/> blank: <input type="text" value="0"/> % <input type="text" value="fx"/> ×
<input type="text" value="Address"/>	Street Address	blank: <input type="text" value="0"/> % <input type="text" value="fx"/> ×
<input type="text" value="Birthday"/>	Date	3/23/2016 to 3/23/2017 in dd/mm/yyyy blank: <input type="text" value="0"/> % <input type="text" value="fx"/> ×
<input type="button" value="Add another field"/>		

Rows: Format: Table Name: ☐ include create table

Sau khi bấm **Generate Data**, các em sẽ được tải về 1 file sql như sau, và chỉ cần copy file này vào SQL Server, chạy là xong.

mysql - HOAN_SS-master (sa [51]) >

Age	Address	Birthday	values	1	'Larry Fuller'	'lfuller0'	'15Y5FH0-1aK'	'lfuller0@aboutads.info'	48	null	'23/10/2014'
Age	Address	Birthday	values	2	'Todd Grant'	'tgrant1'	'pfz127n'	'tgrant1@phinn.com'	88	null	'07/01/2015'
Age	Address	Birthday	values	3	'Banda Campbell'	'ecampbell2'	'6c002CmVA'	'ecampbell2@fastcompany.com'	85	'78822 Hefferne	
Age	Address	Birthday	values	4	'Nicole Tucker'	'ntucker3'	'338ruoXf'	'ntucker3@bizjournals.com'	79	null	'18/03/2014'
Age	Address	Birthday	values	5	'Nancy Franklin'	'nfranklin4'	'uLSXp0iVVF'	'nfranklin4@billionist.com'	42	null	'11/12/201
Age	Address	Birthday	values	6	'Mildred Murphy'	'mmurphy5'	'01kEaC'	'mmurphy5@trellian.com'	10	'3 Northridge Pass	'09/
Age	Address	Birthday	values	7	'Samuel Hudson'	'shudson6'	'shybf7Z'	'shudson6@fda.gov'	11	'01 Marcy Point'	'15/08/2014'
Age	Address	Birthday	values	8	'Douglas Ramirez'	'dramirez7'	'iVhIRnQVul'	'dramirez7@villapages.com'	87	'9 Huxley Str	
Age	Address	Birthday	values	9	'Judith Spencer'	'jspencer8'	'YtK40A'	'jspencer8@arstechnica.com'	89	null	'09/03/2014'
Age	Address	Birthday	values	10	'Ruby Carr'	'rcarr9'	'tth9hLeOn'	'rcarr9@comcast.net'	8	'8 Melvin Alley'	'04/05/2014'
Age	Address	Birthday	values	11	'Tina Tucker'	'ttucker'	'mccaoF1C56'	'ttucker@ucis.edu'	46	'81671 West Pass'	'07/05/2
Age	Address	Birthday	values	12	'Brandon Cole'	'bcole'	'aDEF3aZ170'	'bcole@bisc.gov'	53	'36 Butteridge Junction'	'12/1
Age	Address	Birthday	values	13	'Carolyn Day'	'cday'	'VdKMNbG'	'cday@exer-blog.com'	54	null	'09/06/2014'
Age	Address	Birthday	values	14	'Martin Patterson'	'mpatterson'	'av110B1V'	'mpatterson@aniture.com'	18	null	'29/07/
Age	Address	Birthday	values	15	'Karen Schmidt'	'kschmidt'	'P7Bcd4SA'	'kschmidt@go.gl'	69	null	'31/10/2014'
Age	Address	Birthday	values	16	'Elizabeth Crawford'	'ecrawford'	'bV82sm'	'ecrawford@fifeng.com'	80	null	'04/05/2014'
Age	Address	Birthday	values	17	'James Richardson'	'jrichardson'	'sAOxNT030'	'jrichardson@telegraph.co.uk'	78	'0 Hosp	
Age	Address	Birthday	values	18	'Thomas Burton'	'tburton'	'2exB10'	'tburton@opensource.org'	10	null	'18/09/2014'
Age	Address	Birthday	values	19	'Louise Carroll'	'lcarroll'	'7LmgfY3Ou'	'lcarroll@illinois.edu'	29	'18 Iowa Avenue'	
Age	Address	Birthday	values	20	'Patricia Martin'	'pmartin'	'zFR3PK'	'pmartin@livejournal.com'	90	'85 Stephen Alley'	
Age	Address	Birthday	values	21	'Katherine Barry'	'kbarry'	'B11512'	'kbarry@flickr.com'	95	'4702 Tony Hill'	'17/06/201
Age	Address	Birthday	values	22	'Jeremy Torres'	'jtorres'	'u0CMr'	'jtorres@bualois.com'	90	'0679 Del Mar Road'	'21
Age	Address	Birthday	values	23	'Carlos Hall'	'chall'	'P0109X5'	'chall@e0.net'	65	'7675 Novick Junction'	'20/06/2014'
Age	Address	Birthday	values	24	'Carl Lawson'	'clawson'	'ebu0Ug13'	'clawson@deliciousdays.com'	2	'6 Crescent Oaks Road	
Age	Address	Birthday	values	25	'Jason Jordan'	'jjordan'	'gMPSaYR'	'jjordan@earthwatch.com'	43	'688 Westcott Street	
Age	Address	Birthday	values	26	'Brian Wilson'	'bwilson'	'W3x767vR'	'bwilson@day.com'	27	'1389 Old Shore Place'	'30
Age	Address	Birthday	values	27	'Stechanie Welch'	'swelch'	'WPKiBcB0J'	'swelch@camp.org'	20	null	'06/12/2014'
Age	Address	Birthday	values	28	'Matthew Ramirez'	'mramirez'	'n8Ba61'	'mramirez@ai.com'	49	'5681 Nova Road'	'15/12/
Age	Address	Birthday	values	29	'Kelly Peters'	'kpeters'	'vZ1t0h'	'kpeters@usda.gov'	89	null	'25/02/2014'
Age	Address	Birthday	values	30	'Patricia Woods'	'pwoods'	'chM0FNk0Bd'	'pwoods@google.ca'	89	'05 Crozier Pass'	'17/
Age	Address	Birthday	values	31	'Judy Coleman'	'jcoleman'	'kzR7NpG'	'jcoleman@hugoboss.com'	88	'6 Victoria Point'	
Age	Address	Birthday	values	32	'Sharon Barnes'	'sbarnes'	'UzocP17'	'sbarnes@hicolora.com'	6	null	'02/04/2014'
Age	Address	Birthday	values	33	'Virginia Fields'	'vfields'	'2U421'	'vfields@geocities.com'	54	null	'02/09/2014'
Age	Address	Birthday	values	34	'Raymond Harvey'	'rharvey'	'1B053c'	'rharvey@tiscy.com'	36	null	'13/08/2014'
Age	Address	Birthday	values	35	'Nicholas Rivera'	'nrivera'	'Qv88Jd1H53D'	'nrivera@vistarint.com'	59	'982 Burtan Pl	
Age	Address	Birthday	values	36	'Sandra Gonzales'	'sgonzales'	'CPPa6P'	'sgonzales@iathis.com'	83	'478 Eagle Crest Pl'	
Age	Address	Birthday	values	37	'Betty Tucker'	'btucker0'	'j1x0kV'	'btucker0@com.com'	90	null	'18/08/2014'
Age	Address	Birthday	values	38	'Kathleen Bishop'	'kbishop1'	'4Gvma91'	'kbishop1@kickstarter.com'	52	null	'09/12/2014
Age	Address	Birthday	values	39	'Ruby Rose'	'rrose12'	'Nuu3Cz08se'	'rrose12@unisco.org'	98	'8962 Katie Pass'	'08/07/201
Age	Address	Birthday	values	40	'Todd Robinson'	'trobinson19'	'1SyRd12'	'trobinson19@angelica.jp'	97	'83572 Meadow Ridge	
Age	Address	Birthday	values	41	'Elizabeth Russell'	'erussell11'	'7YozvY1s'	'erussell11@irishin.com'	78	null	'27/02/2003

2.2. Tạo dữ liệu dump data trong source code

2.2.1. Tạo dữ liệu dump data trong javascript

Khi sử dụng một số java script framework, chúng ta sẽ cần 1 số data giả dưới dạng java script object để hiển thị. Ta có thể sử dụng faker.js cho mục đích trên (tải faker.js tại đây (bấm DownloadZIP): <https://github.com/marak/Faker.js/>

Giải nén ra, trong thư mục build lấy file **faker.js**, sau đó thêm reference tới file đó như sau:

```

1 <html lang="en" xmlns="http://www.w3.org/1999/xhtml">
2   <head>
3     <script src="faker.js" type="text/javascript"></script>
4     <title>Test Random</title>
5   </head>
6   <body></body>
7 </html>

```

Việc sử dụng code khá đơn giản, các bạn chỉ cần sử dụng các function trong class **faker**. Tham khảo thêm API ở đây: <http://marak.com/faker.js/>

Dưới đây là code mẫu để tạo 100 object của class Student

```

1 function GetRandomStudent () {

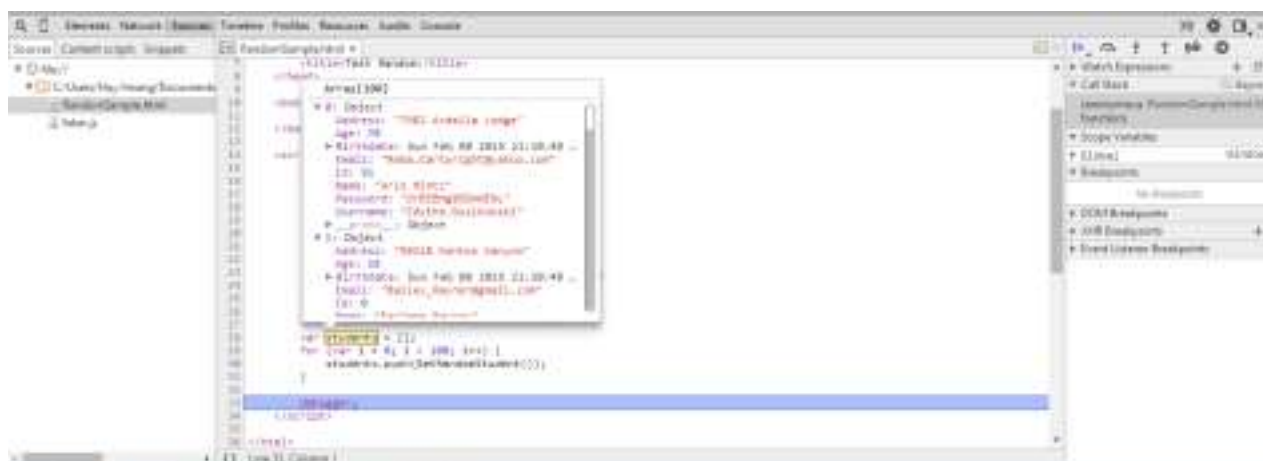
```

```

2      return {
3          Id : faker.random.number(100),
4          Name : faker.name.findName(),
5          Username : faker.internet.userName(),
6          Password: faker.internet.password(),
7          Email : faker.internet.email(),
8          Age : faker.random.number(100),
9          Address: faker.address.streetAddress(),
10         Birthdate: faker.date.past()
11     };
12 }
13 var students = [];
14 for (var i = 0; i < 100; i++) {
15     students.push(GetRandomStudent());
16 }

```

Và đây là kết quả:



2.2.2. Tạo data giả ở server side (trong C#)

Trong một vài trường hợp khác, chúng ta sẽ cần tạo data giả ở server side. Trong ví dụ dưới đây, chúng ta sẽ demo với C#, sử dụng thư viện **Faker.NET** (các ngôn ngữ khác như Java, Ruby, ... có thể tìm thư viện tương ứng).

Các bạn sử dụng Nuget để cài đặt **Faker.NET**.

Mở Nuget Package Manager Console và gõ vào **"Install-Package Faker.Net"** là xong.



Cách sử dụng cũng tương tự như phiên bản javascript, có điều thư viện này chưa hỗ trợ password và datetime.

```

1      public class Student
2      {
3          public int Id { get; set; }
4          public string Name { get; set; }
5          public string Address { get; set; }
6          public string Username { get; set; }
7          public int Age { get; set; }
8          public string Email { get; set; }
9          public static Student GetRandomStudent()
10         {
11             var name = Faker.Name.FullName(NameFormats.Standard);
12             return new Student
13             {
14                 Id = Faker.RandomNumber.Next(100),
15                 Name = name,
16                 Username = Faker.Internet.UserName(name),
17                 Email = Faker.Internet.Email(name),
18                 Age = Faker.RandomNumber.Next(50),
19                 Address = Faker.Address.StreetAddress(),
20             };
21         }
22         //Tạo 100 random student
23         var students = new List<Student>();
24         for (int i = 0; i < 100; i++) students.Add(Student.GetRandomStudent());

```

Và đây là kết quả

Locals			
Name	Value	Type	
args	(string[0])	string[]	
students	Count = 150	System.I	
[0]	(TestSample.Student)	TestSam	
Address	"0126 Matt Mount"	string	
Age	13	int	
Email	"sunny.johns@wunsch.us"	string	
Id	74	int	
Name	"Sunny Johns"	string	
Username	"sunny_johns"	string	
[1]	(TestSample.Student)	TestSam	
Address	"8786 Herminio Dam"	string	
Age	36	int	
Email	"marcelina.krajcik@wisock.biz"	string	
Id	35	int	
Name	"Marcelina Krajcik"	string	
Username	"marcelina_krajcik"	string	
[2]	(TestSample.Student)	TestSam	
Address	"145 Clotilde Stream"	string	
Age	32	int	
Email	"wade_walsh@wolf.info"	string	
Id	34	int	
Name	"Wade Walsh"	string	
Username	"wade.walsh"	string	
[3]	(TestSample.Student)	TestSam	
[4]	(TestSample.Student)	TestSam	
[5]	(TestSample.Student)	TestSam	
[6]	(TestSample.Student)	TestSam	
[7]	(TestSample.Student)	TestSam	