

**A Technical Magazine**  
**2018 - 2019**



# DEXAR

**VOL 3 | ISSUE 1**

**EDITORIAL  
BOARD**

**FACULTY MEMBER**  
**MR. P. N. RAMESH**



## STUDENT MEMBERS

**MR.NAVEEN KUMAR.M.N (IV-CSE)**

**Ms.TEJASWHINI.C.S (IV-CSE)**

**MR.MOHAMED THAKSHIN.S (III-CSE)**

**Ms.JANANI.M (III-CSE)**

**MR.NISHANTH.P (II-CSE)**



# Table of Contents

1. Apple is attempting to sabotage web Technology	-Page 01
2. Artificial Neural Networks Back-propagation Algorithm	-Page 04
3. Artificial intelligence, machine learning, and Deep learning: A Comparison	-Page 06
4. 'DO IT LEARN IT'- ACTIVITY	
➤ ARMSTRONG NUMBER 1	-Page 07
➤ Fibonacci Series 2	-Page 08
➤ Diamond II	-Page 09
5. ANNDotNET's Export Options	-Page 10
6. The development of a custom virtual path provider in ASP.Net 2.0	-Page 12
7. Server less Computing-How does "Server-less Computing" work?	-Page 14
8. SOLUTION FOR 'DO IT LEARN IT' ACTIVITY	
➤ ARMSTRONG NUMBER 1	-Page 16
➤ Fibonacci Series 2	-Page 17
➤ Diamond II	-Page 18



## **Mr.NAVEEN KUMAR.M.N (B.E IV CSE)**

**A few years ago, Mark Reinhold, Chief Architect of the Java Platform Group at Oracle, wrote in a blog post titled "Moving Java ForwardFaster" that Java needed to advance more quickly in order to become competitive.**

A Java framework is specific to the Java programming language and serves as a platform for creating Java applications and web applications. The aim of frameworks is to provide a common structure so that developers don't have to redo it from scratch and can reuse the code provided. It also allows designers and developers to focus on creating unique features for their web-based projects rather than re-inventing the wheel through coding. Spring is one of the most widely-used Java frameworks primarily for the development of web applications. Spring supports such things as application events and listeners, externalized configuration, YAML, and type-safe configuration. Micro service frameworks can be used for deploying Java. Spring Boot is probably the best Microservice in the Java framework that works on top languages for Inversion of Control, Aspect-Oriented Programming, etc

JavaScript (js) is a light-weight object-oriented programming language. JavaScript is so popular that it's the most used programming language in the world, used as a client-side programming language by 98.0% of all websites. Apache Parquet is an open source, column-oriented data file format designed for efficient data storage and retrieval. It has seen widespread adoption for fast analytical querying. The required skill for the upcoming years is to understand the basics of Java frameworks, JavaScript and microservices can enrich the developing knowledge of each and everyone.

**Editorial Student Member : Mr.NAVEEN KUMAR.M.N (B.E IV CSE)**

# Apple is attempting to sabotage web technology



Apps created using Electron are now being silently rejected by the Mac App Store. All of the apps created with Electron may be built using the web-based code. This group includes some of the most well-liked apps available in the App Store, including WhatsApp, Slack, Spotify, and Discord. For developers that require access to these exclusive API functions, Apple doesn't offer any viable alternatives. On its platforms, Apple has a history of stalling the development of the web.

Apple forbids fully independent third-party browsers on iOS, forcing all apps to use the Safari browser when displaying web content. Additionally, by only partially adopting the Progressive Web Apps (PWA) standard, Apple made it too unstable to be relied upon.

The Electron framework, which enables programmers to create web-based programs for Apple platforms without a variety of distinct codebases, is not accepted by the App Store. Some claim that this

causes lower-quality apps to be created, but I would counter that the alternative is to create no apps at all or apps that are infrequently updated. Apple now released a rival framework called Catalyst that enables iPad app makers to swiftly distribute their products to Mac users.



## CLR functions for SQL Server

Today, we had to use SQL to perform some fairly odd tasks that involved using an encryption library in SQL server. I had never done this before, so I figured I'd blog about it.

There are several steps involved:

1. Create the CLR DLL for the SQL function to use, and copy it to SQL Binn
2. Register the CLR DLL in SQL Server
3. Create a normal SQL function that uses the CLR DLL

The first part is straight forward enough, the following code gives an example:

```

C# Shrink ▲ Copy Code
1: using System;
2: using System.Collections.Generic;
3: using System.Text;
4: using Microsoft.SqlServer.Server;
5: using System.Data;
6: using System.Data.Sql;
7: using System.Data.SqlTypes;
8: using Encrypt;
9:
10: public class StoredProcedures
11: {
12:
13:     [Microsoft.SqlServer.Server.SqlFunction()]
14:     public static string Enc(SqlString password, SqlString encStringOut)
15:     {
16:         Encryption enc = new Encryption();
17:         return enc.Encrypt(password.ToString(), encStringOut.ToString());
18:     }
19:
20:     [Microsoft.SqlServer.Server.SqlFunction()]
21:     public static string Dec(SqlString password, SqlString encStringOut)
22:     {
23:         Encryption enc = new Encryption();
24:         return enc.Decrypt(password.ToString(), encStringOut.ToString());
25:     }
26: }

```

So that's easy enough. Compile this job done.

We must now work on the SQL server. I therefore copied the SQL Server Encryption first. Dll to the SQL server computer's

**C:\Program Files\MicrosoftSQLServer\MSSQL.1\MSSQL\Binn directory.**

In order to make the DLL registration code that needs to be performed for SQL a little bit easier, I also transferred the DLL produced (SQLServerEncryption.Dll) to the C: drive on the SQL server machine.

Since we have already copied to \binn and C:\, we must now register the DLL with SQL server. Let's examine that now. First and foremost, the SQL server installation has to support CLR types. Which either involves using the SQL listed below:

```

C#
EXEC dbo.sp_configure 'clr enabled', 1 RECONFIGURE WITH

```

OR

If you experience difficulties doing it that way, utilise the "Surface Area Configuration For Features" link in the SQL Server Surface Area Configuration and then select CLR integration from the displayed treeview. Following completion of this, the CLR DLL can be registered with SQL as follows:

```

C# Copy Code
create assembly SQLServerEncryption from 'c:\SQLServerEncryption.dll' WITH PERMISSION SET = SAFE

```

After completing that, all that remains is to write a typical SQL server function that makes use of the CLR DLL. It is easily accomplished as follows:

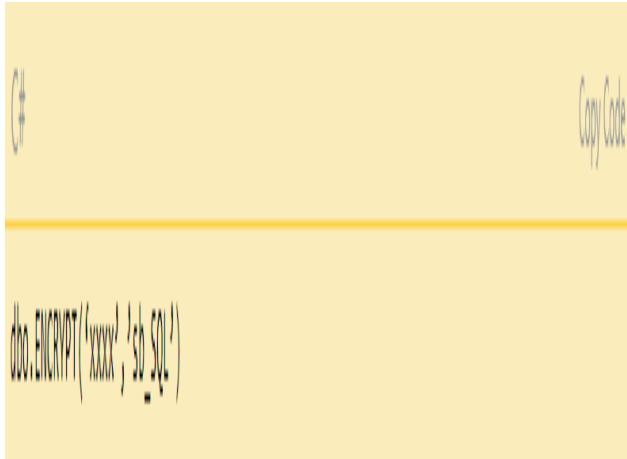
```

C# Copy Code
ALTER FUNCTION [dbo].[ENCRYPT](@password [nvarchar](255), @encStringOut [nvarchar](255))
RETURNS [nvarchar](255) WITH EXECUTE AS CALLER
AS
EXTERNAL NAME [SQLServerEncryption].[StoredProcedures].[Enc]

```

That's all, then! The CLR Function is now available for use as desired.

**For example:**

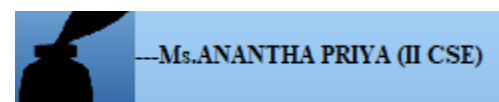


```
ScalarFunctions.cs * X
CLR_ScalarFunctions.sqlproj UserDefinedFunctions
1 using System;
2 using System.Data;
3 using System.Data.SqlClient;
4 using System.Data.SqlTypes;
5 using Microsoft.SqlServer.Server;
6
7 public partial class UserDefinedFunctions
8 {
9     [Microsoft.SqlServer.Server.SqlFunction]
10    public static SqlString ScalarFunctions()
11    {
12        // Put your code here
13        return new SqlString (string.Empty);
14    }
15 }
16
```

## Conclusion:

So Apple, which claims to use the EPUB format exclusively, has now created an incompatible, proprietary version of that format. And with iBooks Author they've added licensing terms that restrict what an author can do with the generated content.

In that scenario, I as a publisher would have had multiple choices. My readers would have had multiple choices. Apple would have had the opportunity to prove in the marketplace that its new format was a superior option to the industry-leading standard. Why buy a boring old e-book when you can get an exciting interactive experience?



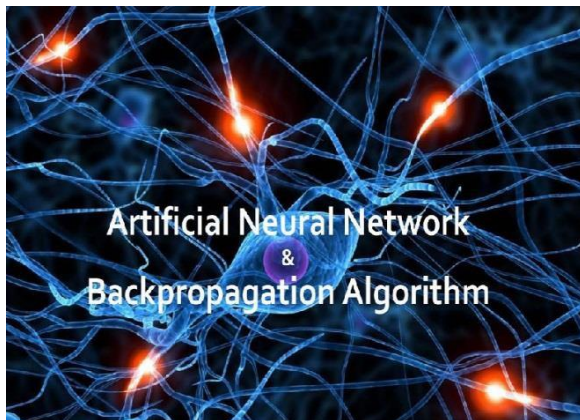


# ARTIFICIAL NEURAL NETWORKS' BACKPROPAGATION ALGORITHM

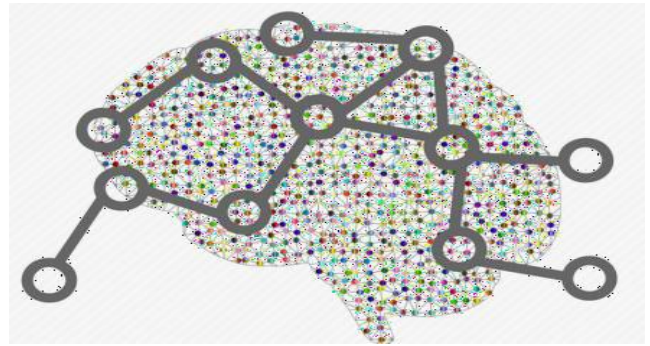
We will discover back propagation-related information in this article, which will help you better understand the topic of ANNs' learning mechanisms overall.

## Cost Function Assumptions

In order to make this article easier to understand, from now on, we are going to use specific cost function – we are going to use quadratic cost function, or mean squared error function:



As mentioned, there are some assumptions that we need to make regarding this function in order for backpropagation to be applicable.



To be exact, there are two of them:

1. The cost function can be written as an average:

$$C = \frac{1}{n} \sum_x C_x$$

2. The cost function can be written as a function of the outputs from the artificial neural network.

$$C(w, b) \equiv \frac{1}{2n} \sum_x \|y(x) - a\|^2.$$

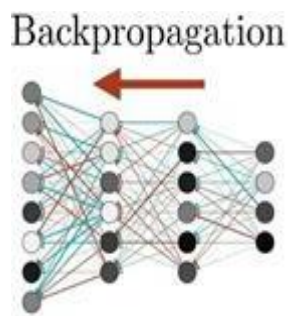
## Backpropagation Algorithm

We already established that backpropagation helps us understand how changing the weights and biases affects the cost function. This is achieved by calculating partial derivatives for each weight and for each bias, i.e.,  $\partial C / \partial w$  and  $\partial C / \partial b$ . Now, every neuron in the neural network generates some sort of an error. This error affects other neurons and ultimately it affects the

global error, meaning it affects our cost function. The middle step of this whole process is calculating this value, and use it to align weights accordingly. By calculating partial derivate of cost function by the input of each neuron, we define the error:

$$\delta = \frac{\partial C}{\partial net}$$

where *net* is the weighted input in the certain neuron.



We took a slight detour, didn't we? Let's examine this algorithm's primary steps. Backpropagation essentially consists of two stages.

Propagation is the first, and it consists of the following steps:

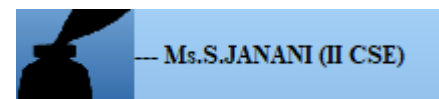
1. Initialize weights of neural network
2. Propagate inputs forward through the network to generate the output values
3. Calculate the error
4. Propagation of the output back through the network in order to generate the error of all output and hidden neurons

The second part of backpropagation updates weights of connections:

The weight's output error and input are multiplied to find the gradient of the weight. A certain percentage, defined by learning rate (more on this a bit later) of the weight's gradient is subtracted from the weight.

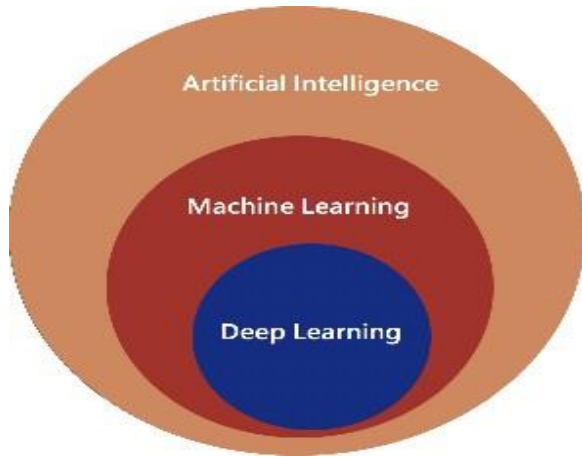
## Conclusion:

The technology known as backpropagation has a significant impact on the field of artificial neural networks. It improved the entire weight updating procedure and, in a way, aided the growth of this industry. Without delving into the mathematics underlying this technique, building an artificial neural network is possible using one of the robust libraries now available; nonetheless, knowing the mathematics underlying this approach is invaluable.





# Artificial intelligence, machine learning, and deep learning: A Comparison



The term "Artificial intelligence" can be confused with other terms like "Machine Learning" and "Deep Learning". AI and the Internet of Things are inextricably intertwined. Several technological advances are converging at once to set the foundation for an AI and IoT explosion.

## So what's the difference between AI, ML, and DL?

### AI

The term artificial intelligence (AI) was coined in 1956 by John McCarthy. AI involves machines that can perform tasks characteristic of human intelligence. These include planning, understanding language, recognizing objects and sounds, learning, and problem-solving. A machine that's great at recognizing images, but

nothing else, would be an example of narrow AI.

### Machine Learning

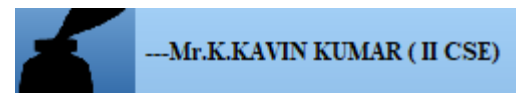
At its core, machine learning is simply a way of achieving AI.

Artificial intelligence (AI) is the ability to learn without being programmed. Arthur Samuel coined the phrase not too long after AI, in 1959. Machine learning is a way of "training" an algorithm so that it can learn how to perform tasks. "Training" involves feeding huge amounts of data to the algorithm and allowing the algorithm to adjust itself and improve. Machine learning is the ability of a machine to learn what an object looks like by studying millions of images and videos.

### Deep Learning



It is one of many approaches to machine learning, and can be applied to any number of human-computer interactional systems. Deep learning was inspired by the structure and function of the brain, namely the interconnecting of many neurons. In ANNs, there are "neurons" which have discrete layers and connections to each other. Each layer picks out a specific feature to learn, such as curves/edges in image recognition.



## ‘DO IT LEARN IT’- ACTIVITY

### ARMSTRONG NUMBER 1

**Problem Description:**

Get an Integer n and Find whether the n is ArmStrong or Not ArmStrong

**Input Format:**

Integer

**Constraints**

$0 \leq n \leq 100000000$

**Output Format:**

String

**Sample Input:**

153

**Sample Output:**

armstrong

**Explanation:**

13+53+33 is not equal to 153

**For Solutions Refer Page No:18**

## Fibonacci Series 2

### Problem Description:

Get two integers n and m. print the Fibonacci numbers which lies between n and m

### Input Format:

Two Integers

### Constraints:

$0 \leq n \leq 1000$

$0 \leq m \leq 1000$

$n \leq m$

### Output Format:

Fibonacci Series

### Sample Input:

5 20

### Sample Output:

5 8 13

### Explanation:

0 1 1 2 3 5 8 13 21 - in this only 5 8 13 lies in between 5 and 20

**For Solutions Refer Page No:19**

## Diamond II

### Problem Description:

Get an Integer N and print the following Diamond for N Rows

### Input Format:

Integer N

### Constraints:

$0 \leq N \leq 100$

### Output Format:

Pattern

### Sample Input:

4

### Sample Output:

```
1
121
12321
1234321
12321
121
1
```

### Explanation

Diamond Pattern

**For Solutions Refer Page No:20**

# ANNdotNET's Export Options

## Introduction

Recently, ANNdotNET v1.0 was released, and the response has been very good. Additionally, the release hasn't yet contained any severe bugs or blocking issues, which makes me very delighted. In this blog article, we'll look at ANNdotNET's export choices.

## About ANNdotNET

The ANNdotNET is designed to be an application that can provide the entire life-cycle for machine learning projects, including the definition of the raw data collection, cleaning and engineering of features, as well as training and model evaluation.

Currently, ANNdotNET defines three export options:

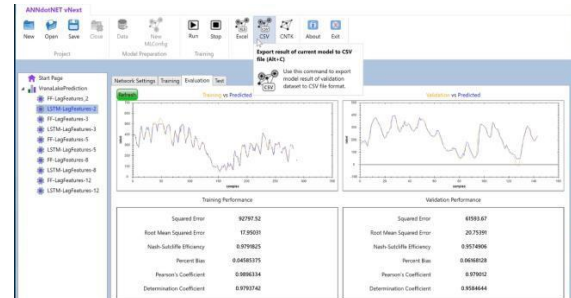
1. Export model result to CSV file,
2. Export model and model result to Excel, and
3. Export model in CNTK file format.

We can achieve a wide variety of ML scenarios using those three output options.

## Export to CSV

Actual and anticipated values from the testing data set can be exported to a comma separated text file using the export to CSV option. In the absence of a testing data set, the results of the validation data set will be exported. The export procedure is stopped if neither a testing dataset nor a validation dataset is provided.

The proper mlconfig file is chosen to begin the export procedure. Before being exported, the network model needs to be trained.



The csv file is produced on disc once the export procedure is finished. The exported output can be imported into Excel, where comparable content will be displayed as illustrated in the figure below:

The screenshot shows an Excel spreadsheet with two columns: 'Hsr\_actual' and 'Hsr\_predicted'. The data is as follows:

	A	B	C
1	Hsr_actual	Hsr_predicted	
2	288	245.9566	
3	321	318.691	
4	314	335.6852	
5	299	302.9023	
6	274	288.5725	
7	256	253.8912	
8	252	246.6156	
9	291	262.8735	
10	307	308.806	

The actual and predicted values are displayed in two columns in the exported result.

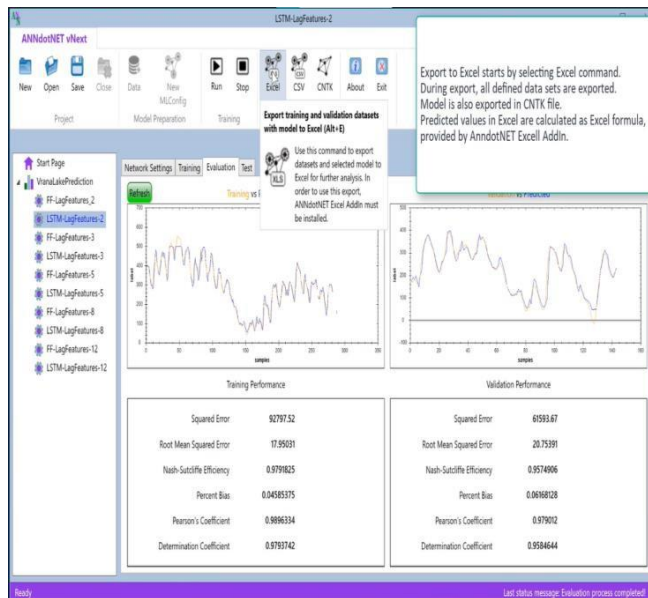
## Export to EXCEL

The Excel export option does more than just export the outcome. In actuality, the model is being deployed into an Excel environment. The model is exported in addition to all defined datasets (training,



validation, and test). The model evaluation refers to the ANNdotNET Excel Add-in, which is used to produce predicted values, as an ordinary Excel formula. You can continue the analysis of the model and associated datasets by opening the exported xlsx file. The exported model for the Concrete Slum Test example is shown in the following image. Only the training and validation datasets are defined, therefore they are exported.

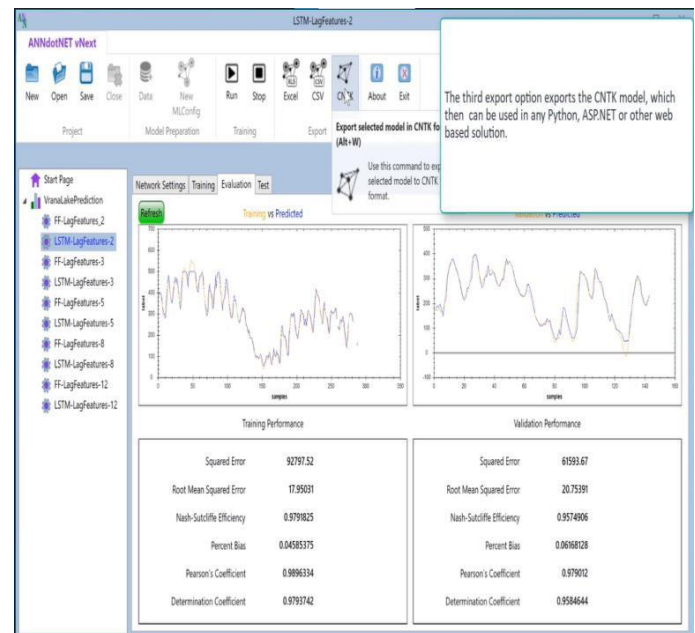
1	Test Data			
2	Hsr-2	Hsr-1	Hsr_actual	Hsr_predicted
3	213	224	288	ANNdotNETEval(A3:B3, "C:\Users\bhnrjica\Desktop\vrana_model_2_lag.xlsx.model")
4	224	288	321	
5	288	321	314	
6	321	314	299	
7	314	299	274	
8	299	274	256	
9	274	256	252	
10	256	252	291	
11	252	291	307	
12	291	307	306	



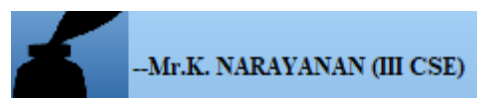
Once the formula has been evaluated for the first row, we can use Excel trick to copy it in the other rows. The same thing applies to other data sets that have been divided into Excel Worksheets.

## Export to CNTK

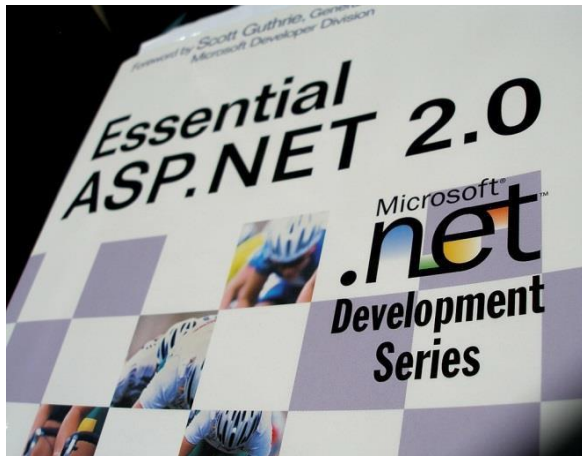
The final choice enables exporting of trained CNTK models in CNTK format.



It simply export into \*.cntk model. It can be used in any Python, ASP.NET or other web based solutions.



# The development of custom virtual path provider in ASP.NET 2.0



## Introduction

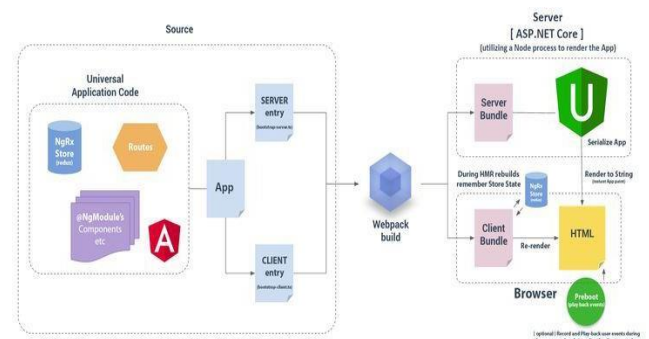
Virtual path provider in ASP.NET 2.0 is a way to make the world believe that something exists on the server which in reality is not. The web application can now access resources stored in a virtual file system thanks to this. This enables the virtualization of file systems containing browseable resources. Virtual file system can contain any file in the web application other than

1. Binaries
2. Web.config
3. Global.asax
4. Files in special folder such as App\_Data and so on

Since Asp.net loads all of the mentioned resources before the virtual path provider, they are unable to be seen inside the virtual file system. Additionally, the content inside the virtual file system (the content supplied by the virtualpathprovider) is not compiled if the site is precompiled.

## Creating a custom Virtual Path Provider

We need that to specify how to relate your virtual file with the request. The task of the Custom VirtualPathProvider is to translate the user's request for the Virtual File into a form that ASP.Net can understand. To begin with, we must develop a class by deriving from the base VirtualPathProvider class. Although the basic class for VirtualPathProvider supports a number of methods, it is not specifically designed for any circumstance. Therefore, we must override the methods that acquire the necessary data for us, process the data (if necessary), and send the processed markup to the ASP.Net.



```

C# Copy Code
public class TextVirtualPathProvider : VirtualPathProvider
{
    public static void AppInitialize()
    {
        ...
    }

    public override bool FileExists( string virtualPath )
    {
        ...
    }

    public override VirtualFile GetFile( string virtualPath )
    {
        ...
    }
}

```

The code above specifies a custom processing logic for `TextVirtualPathProvider`, which inherits from `VirtualPathProvider` and overrides the methods `FileExists()` and `GetFile()`.

`AppInitialize()` is a special method. This function, which is called before anything else in this class, functions somewhat like `Main()` for the `VirtualPathProvider`.

This method often specifies an alternative execution path by registering this `VirtualPathProvider` with the ASP.Net Compilation system.

```

C# Copy Code
HostingEnvironment.RegisterVirtualPathProvider(new TextVirtualPathProvider());

```

In some `CustomVirtualPathProviders`, this function is optional. Even in the `Global.asax` file, we can specify the same code inside the `Application Start()`. The `GetFile()` method is an additional crucial one. The kind of `VirtualFile` returned by this procedure. The ASP.Net engine uses this method to retrieve the markup for the virtual file. The next stage

in developing the custom provider is to develop a class that derives from the `VirtualFile` class and understands precisely what must be done in response to a request for a virtual file.

```

C# Copy Code
public class TextFile : VirtualFile
{
    public TextFile( string virtualPath, TextVirtualPathProvider provider )
        : base( virtualPath )
    {
        ...
    }

    public override Stream Open()
    {
        ...
    }
}

```

This class is pretty straightforward as well, a constructor and an override method. When the handle to this object is obtained, the ASP.Net engine calls the `Open()` function to obtain the stream containing the markup. Therefore, the primary method in the entire code that must comprehend the virtual file system and know how to handle any requests for virtual files is `Open()`.



# Server less Computing

## How does "Serverless Computing" work?

When I first heard the term "Serverless Computing," I was terrified, wondering how this could possibly be practical.



Actually, "Serverless Computing" is a misnomer. Even though it says serverless computing, a server is still necessary. The difference is that we are not required to independently maintain a physical server. A server is kept in a cloud, and all we need to do to control it is give the cloud the right instructions. These directives are typically transmitted to the cloud utilising functions. In 2014, Amazon debuted "AWS Lambda," the first public cloud to offer serverless computing. Recently, "Google Firebase" has added serverless capabilities to its list of cloud services. Developers can upload functions containing their back-end code to the cloud, where the cloud will automatically execute the necessary logic in response to event triggers

and HTTP requests. Now, let's practise serverless computing using Google Firebase, which is available for learners at no cost

We have to installed Node.js

### 1. First, install the Firebase CLI using the following command.

```
npm install -g firebase-tools
```

### 2. Authentication & Initialization

```
firebase login
```



Then run the following commands to authenticate and initialize the project. You will need to have a google account for authentication. The browser will open when this command is executed, and you'll be taken to a login page. When everything is complete, enter the credentials, then head to the Firebase console to start a new project. This



will be synchronised with the local project when it is launched.

### 3. Importing Modules

After configuration is complete, go to the project folder and you'll find that an `index.js` file has been generated. In order to create functions and configure triggers, we must first import the Cloud Functions. The Firebase Real-Time Database must then be accessed by the Admin SDK in order to save our input.



## 4. Adding our First Functions

```

1 exports.saveInput = functions.https.onRequest((req, res) => {
2   // Get the 'text' parameter.
3   const original = req.query.text;
4   // Save the text
5   return admin.database().ref('/messages').push({original: original}).then((snapshot) => {
6     // Redirect with 303 SEE OTHER to the URL of the pushed object in the Firebase console.
7     return res.redirect(303, snapshot.ref);
8   });
9 });

```

There aren't many differences between this and Node.js, if you're familiar with it. On essence, we are collecting the request input and saving it to the database via the admin module in the /message path. Every text we

save will be kept in the real-time database under messages. When we run the code, the Firebase dashboard will show you this.

## 5. Adding our Second Functions

We can retrieve the event's original value because this function is event-triggered. After that, we change the case to uppercase and save it there. You'll understand this once we execute and view the outcomes in the Firebase dashboard.

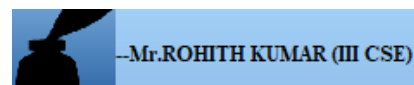
## 6. Deploy Function

Now we have implemented our functions and it's time to deploy them to cloud. Run the following command to deploy our functions.



Finally it will deploy each of our routines in turn while providing a URL to activate our HTTP-triggered function.

I hope this post gave you a foundational understanding. This is a very simple illustration to help explain how to utilise Google Cloud Functions.





## SOLUTION FOR 'DO IT LEARN IT' ACTIVITY

### ARMSTRONG NUMBER 1

#### PROGRAM :

```
#include <stdio.h>
#include <string.h>
#include <math.h>
#include <stdlib.h>
int main()
{
    int num, Num, rem, result = 0;

    scanf("%d", &num);
    Num = num;

    while (Num != 0) {

        rem = Num % 10;

        result += rem * rem * rem;

        Num /= 10;
    }

    if (result == num)
        printf("armstrong");
    else
        printf("not armstrong");

    /* Enter your code here. Read input from STDIN. Print output to STDOUT */
    return 0;
}
```

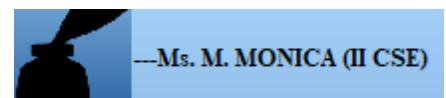


## Fibonacci Series 2

### PROGRAM :

```
#include <stdio.h>
#include <string.h>
#include <math.h>
#include <stdlib.h>

int main() {
    int n,m,t1=0,t2=1,t3=0;
    scanf("%d %d",&n,&m);
    while(t3<=m)
    {
        if(t3>=n)
        {
            printf("%d ",t3);
        }
        t1=t2;
        t2=t3;
        t3=t1+t2;
    }
    return 0;
}
```



## Diamond II

### PROGRAM :

```
#include <stdio.h>
#include <string.h>
#include <math.h>
#include <stdlib.h>
int main()
{
    int i,j,n,k,step=1,num;
    scanf("%d",&n);
    for(i=1;i<=n;i++)
    {
        num=1;
        for(k=1;k<=n-i;k++)
        {
            printf(" ");
        }
        for(j=1;j<=step;j++)
        {
            printf("%d",num);
            if(j<i)
            {
                num++;
            }
            else
            {
                num--;
            }
        }
        step+=2;
        printf("\n");
    }
    step=2*n-1;
    int temp=n-1;
    for(i=1;i<n;i++)
    {
        num=1;
        for(j=1;j<=i;j++)
```

```
{
    printf(" ");
}
for(k=1;k<=step-2;k++)
{
    printf("%d",num);
    if(k<temp)
    {
        num++;
    }
    else
    {
        num--;
    }
}
temp--;
step-=2;
printf("\n");
}
return 0;
}
```



---Ms. J.A. KEERTHANA (III CSE)



## **Editorial Faculty Member**

**Mr.P.N.Ramesh**

## **Editorial Student Members**

**Mr.Naveen kumar.M.N (IV-CSE)**

**Ms.Tejaswhini.C.S (IV-CSE)**

**Mr.Mohamed Thakshin.S (III-CSE)**

**Ms.Janani.M (III-CSE)**

**Mr.Nishanth.P (II-CSE)**