

FULLSTACK DOTNET TUTORIAL

In this tutorial, we are going to build a full stack dotnet project, which is connected to a database and an HTML frontend

Let Start With The BACKEND

STEP 1

- ➔ Create the project by using the visual studio code application
- ➔ Click on the [project>Manage nugget packages] to install the following packages
 1. Microsoft.EntityFrameworkCore
 2. Microsoft.EntityFrameworkCore.SqlServer
 3. Microsoft.EntityFrameworkCore.Design
 4. Microsoft.identity.Client
 5. Microsoft.Data.SqlClient
 6. Microsoft.SqlServer.Server
- ➔ To add CORS to your application, create a folder in the project root and name it Extensions
- ➔ Inside the Extensions folder, create a file and name it **ServiceExtensions.cs** and paste this code

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Threading.Tasks;

namespace ProjectManager.Extensions
{
    public static class ServiceExtensions
    {
        public static void ConfigureServices(this IServiceCollection services) =>
        services.AddCors(options =>
        {
            options.AddPolicy("CorsPolicy", builder =>
            builder.AllowAnyOrigin()
            .AllowAnyMethod()
            .AllowAnyHeader());
        });

        public static void ConfigureIISIntegration(this IServiceCollection services) =>
        services.Configure<IISSOptions>(options =>
        {
        });

    }
}

➔ In the Program.cs, add ConfigureServices and ConfigureIISIntegration as a service and also register the CORS in the Program.cs file. In the Program.cs, we need to call the DataContext in it too like this [builder.Services.AddDbContext<DataContext>();]
```

The entire **Program.cs** should look like this

```
using Microsoft.AspNetCore.HttpOverrides;
using ProjectManager.Data;
using ProjectManager.Extensions;

var builder = WebApplication.CreateBuilder(args);

// Add services to the container.

builder.Services.ConfigureCors();
builder.Services.ConfigureIISIntegration();

builder.Services.AddControllers();
// Learn more about configuring Swagger/OpenAPI at https://aka.ms/aspnetcore/swashbuckle
builder.Services.AddEndpointsApiExplorer();
builder.Services.AddSwaggerGen();
builder.Services.AddDbContext<DataContext>();

var app = builder.Build();

if (app.Environment.IsDevelopment())
    app.UseDeveloperExceptionPage();
else
    app.UseHsts();

// Configure the HTTP request pipeline.
if (app.Environment.IsDevelopment())
{
    app.UseSwagger();
    app.UseSwaggerUI();
}

app.UseDefaultFiles();
app.UseStaticFiles();
app.UseForwardedHeaders(new ForwardedHeadersOptions
{
    ForwardedHeaders = ForwardedHeaders.All
});
app.UseCors("CorsPolicy");

app.UseHttpsRedirection();

app.UseAuthorization();

app.MapControllers();
```

```
app.Run();
```

Step 2

In the Root folder, create 2 folders, Models and Data

In the Models folder create this Four Models

Project.cs

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Threading.Tasks;

namespace ProjectManager.Models
{
    public class Project
    {
        public int Id {get; set;}
        public string name {get;set;}
        public string description {get;set;}
        public DateTime dueDate {get;set;}
        public double cost {get;set;}
        public string developer {get;set;}
        public string customer {get;set;}
        public string status {get;set;}
    }
}
```

User.cs

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Threading.Tasks;

namespace FlexiSecure.Models
{
    public class User
    {
        public int Id { get; set; }

        public string Name { get; set; } = string.Empty;
        public string Email { get; set; } = string.Empty;
        public byte[] PasswordHash { get; set; } = new byte[32];
        public byte[] PasswordSalt { get; set; } = new byte[32];
        public string? VerificationToken {get; set; }
        public DateTime? VerifiedAt { get; set; }
    }
}
```

```

public string? PasswordResetToken { get; set; }
public DateTime? ResetTokenExpires{get; set; }

}
}

```

UserRegisterRequest.cs

```

using System;
using System.Collections.Generic;
using System.ComponentModel.DataAnnotations;
using System.Linq;
using System.Threading.Tasks;

namespace FlexiSecure.Models
{
    public class UserRegisterRequest
    {
        [Required,MinLength(3, ErrorMessage ="Enter a valid name")]
        public string Name { get; set; } = string.Empty;
        [Required, EmailAddress]
        public string Email { get; set; } = string.Empty;
        [Required, MinLength(6,ErrorMessage ="Please enter at least 6 characters ")]
        public string Password { get; set; } = string.Empty;
        [Required, Compare("Password")]
        public string ConfirmPassword{get; set; } = string.Empty;

    }
}

```

UserLoginRequest.cs

```

using System;
using System.Collections.Generic;
using System.ComponentModel.DataAnnotations;
using System.Linq;
using System.Threading.Tasks;

namespace FlexiSecure.Models
{
    public class UserLoginRequest
    {

        [Required, EmailAddress]
        public string Email { get; set; } = string.Empty;
        [Required]
        public string Password { get; set; } = string.Empty;

    }
}

```

```
}
```

In the Data folder, Create a **DataContext.cs** file and paste this content

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Threading.Tasks;
using FlexiSecure.Models;
using Microsoft.EntityFrameworkCore;
using ProjectManager.Models;

namespace ProjectManager.Data
{
    public class DataContext : DbContext
    {
        //Empty constructor
        public DataContext(): base(){
        }

        //Database Connection String
        protected override void OnConfiguring(DbContextOptionsBuilder optionsBuilder)
        {
            base.OnConfiguring(optionsBuilder);
            optionsBuilder.UseSqlServer("Server=localhost,1433;Database=Projects;User=sa;Password=HydotTech;TrustServerCertificate=true;");
        }

        //Data Set, where Project and User are models in the Model folder
        public DbSet<Project> Projects => Set<Project>();
        public DbSet<User> Users => Set<User>();

    }
}
```

Step 3

Open the Controller Folder and Create 2 Controllers,

ProjectController.cs

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Threading.Tasks;
using Microsoft.AspNetCore.Mvc;
using Microsoft.EntityFrameworkCore;
using ProjectManager.Data;
using ProjectManager.Models;

namespace ProjectManager.Controllers
{
    [ApiController]
    [Route("api/Projects")]
    public class ProjectController : ControllerBase
    {
        private readonly DataContext _context;
        public ProjectController(DataContext context){
            _context = context;
        }
        [HttpPost()]
        public async Task<IActionResult> AddProject(Project project){
            //Call the Project.cs model to create a skeleton
            var my_project = new Project{
                name = project.name,
                description = project.description,
                dueDate = project.dueDate,
                cost = project.cost,
                developer = project.developer,
                customer = project.customer,
                status = project.status
            };
            _context.Projects.Add(my_project);
            await _context.SaveChangesAsync();
            return Ok("Projects Created Successfully");
        }
        [HttpGet()]
        public async Task<ActionResult<IEnumerable<Project>>> GetItems()
        {
            var items = await _context.Projects.ToListAsync(); // Replace "YourModels" with
            your actual DbSet name

            return Ok(items);
        }
        [HttpGet("{Id}")]
        public async Task<ActionResult<Project>> GetItem(int Id){
```

```

var item = await _context.Projects.FindAsync(Id);
if (item == null){
return NotFound();
}
return Ok(item);

}

```

```

[HttpPut("{id}")]
public async Task<IActionResult> UpdateItem(int id, Project updatedItem){
if (id != updatedItem.Id)
{
return BadRequest("The IDS DONT MATCH"); // Returns HTTP 400 if the ID in the
route does not match the ID in the updated item
}
}

```

```

_context.Entry(updatedItem).State = EntityState.Modified;

```

```

try
{
await _context.SaveChangesAsync();
}
catch(DbUpdateConcurrencyException){
if (!ItemExists(id))
{
return NotFound(); // Returns HTTP 404 if the item with the specified ID is not
found
}
else
{
throw;
}
}
return NoContent();
}
private bool ItemExists(int id)
{
return _context.Projects.Any(e => e.Id == id);
}
}

```

```

[HttpDelete("{id}")]
public async Task<IActionResult> DeleteItem(int id)
{

```

```
var item = await _context.Projects.FindAsync(id);
```

```
if (item == null)
{
    return NotFound("Item not found");
}
```

```
_context.Projects.Remove(item);
await _context.SaveChangesAsync();
```

```
return NoContent();
}
```

```
}
```

```
}
```


UserController.cs

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Security.Cryptography;
using System.Threading.Tasks;
using FlexiSecure.Models;
using Microsoft.AspNetCore.Mvc;
using Microsoft.EntityFrameworkCore;
using ProjectManager.Data;

namespace FlexiSecure.Controllers
{
    [ApiController]
    [Route("api/Auth")]
    public class UserController : ControllerBase
    {
        private readonly DataContext _context;
        public UserController(DataContext context)
        {
            _context = context;
        }

        [HttpPost("register")]
        public async Task<IActionResult> Register(UserRegisterRequest request)
        {
            //Check if Email is already registered
            if(_context.Users.Any(u=> u.Email == request.Email)){
                return BadRequest("The user account is already registered");
            };

            //Create a Password Hash
            CreatePasswordHash(request.Password, out byte[]passwordHash,out
            byte[]passwordSalt);

            //Call the user model from User.cs, create a new objects
            var user = new User{
                Name = request.Name,
                Email = request.Email,
                PasswordHash=passwordHash,
                PasswordSalt = passwordSalt,
                VerificationToken = CreateRandomToken(),
            };

            // The _context is called from the DataContext file and inside that file, we
            have the Users data set
            //The Users dataset is using the User model
        }
    }
}
```

```
//We will the add the user object we created to the Users dataset
_context.Users.Add(user);
await _context.SaveChangesAsync();

return Ok("Hello "+request.Name+", your account has successfully been created");

}
```

```
[HttpPost("login")]
public async Task<IActionResult>Login(UserLoginRequest request)
{

var user = await _context.Users.FirstOrDefaultAsync(u => u.Email ==
request.Email);
if(user==null){
return BadRequest("User not found");
}

if(!VerifyPasswordHash(request.Password, user.PasswordHash, user.PasswordSalt))
{
return BadRequest("Password is incorrect");
}

if(user.VerifiedAt == null){
return BadRequest("User is not verified");
}

return Ok($"Welcome Back, {user.Name}! ");

}
```

```
[HttpPost("verify")]
public async Task<IActionResult>Verify(string token)
{

var user = await _context.Users.FirstOrDefaultAsync(u => u.VerificationToken ==
token);
if(user==null){
return BadRequest("Invalid token");
}

user.VerifiedAt = DateTime.Now;
```

```
await _context.SaveChangesAsync();
```

```
return Ok("Congratulations! You have been verified");
```

```
}
```

```
private static bool VerifyPasswordHash(string password, byte[] passwordHash,  
byte[] passwordSalt)
```

```
{  
    using (var hmac = new HMACSHA512(passwordSalt))  
    {
```

```
        var computedHash = hmac.ComputeHash(System.Text.Encoding.UTF8.GetBytes(password));  
        return computedHash.SequenceEqual(passwordHash);
```

```
    }
```

```
}
```

```
private static void CreatePasswordHash(string password, out byte[] passwordHash,  
out byte[] passwordSalt)
```

```
{  
    using (var hmac = new HMACSHA512())  
    {
```

```
        passwordSalt = hmac.Key;  
        passwordHash = hmac.ComputeHash(System.Text.Encoding.UTF8.GetBytes(password));
```

```
    }
```

```

}

private string CreateRandomToken(){
return Convert.ToHexString(RandomNumberGenerator.GetBytes(64));
}

}
}

```

Bonus Feature

If you want to add forgot password and reset password feature,

1. Create a **ResetPasswordRequest.cs** model and add the following code

```

using System;
using System.Collections.Generic;
using System.ComponentModel.DataAnnotations;
using System.Linq;
using System.Threading.Tasks;

namespace FlexiSecure.Models
{
public class ResetPasswordRequest
{
[Required]
public string Token { get; set; } = string.Empty;
[Required, MinLength(6,ErrorMessage ="Please enter at least 6 characters ")]
public string Password { get; set; } = string.Empty;
[Required, Compare("Password")]
public string ConfirmPassword{get; set; } = string.Empty;

}
}

```

Inside the **UserController.cs**, add the following code

```

[HttpPost("forgot-password")]
public async Task<IActionResult>ForgotPassword(string email)
{

var user = await _context.Users.FirstOrDefaultAsync(u => u.Email == email);
if(user==null){
return BadRequest("User " + email + " does not exist");
}

user.PasswordResetToken = CreateRandomToken();
user.ResetTokenExpires = DateTime.Now.AddDays(1);
await _context.SaveChangesAsync;

return Ok("You may now reset your password");
}

```

```
}
```

```
[HttpPost("reset-password")]
public async Task<IActionResult>ResetPassword(ResetPasswordRequest request)
{

    //The PasswordResetToken which i declared in the User.cs model and it is made
    //availabe in the
    //DataContext.cs through the DbSet should be equal to the Token the user will
    //declare in the ResetPasswordRequest model
    var user = await _context.Users.FirstOrDefaultAsync(u => u.PasswordResetToken ==
    request.Token);
    if(user==null || user.ResetTokenExpires<DateTime.Now ){
    return BadRequest("Invalid Token");
    }

    CreatePasswordHash(request.Password, out byte[] passwordHash, out byte[]
    passwordSalt);
    user.PasswordHash = passwordHash;
    user.PasswordSalt = passwordSalt;
    user.PasswordResetToken = null;
    user.ResetTokenExpires = null;

    await _context.SaveChangesAsync();

    return Ok("Password successfully reset.");
}
```

Step 4

Ensure that the Database is running in Docker or any place. To create a Database in Docker, follow this tutorial

Adding Database

Follow this link to create a database sql server using docker in Mac Os

<https://builtin.com/software-engineering-perspectives/sql-server-management-studio-mac>

Follow the Steps Below

STEP 1: DOWNLOAD AND INSTALL DOCKER

STEP 2: DOWNLOAD THE MS SQL SERVER IMAGE TO DOCKER

Open the terminal in the Mac and run the following code. This will install the 2022 MS SQL Server image on your device.

```
sudo docker pull mcr.microsoft.com/mssql/server:2022-latest
```

STEP 3: LAUNCH THE SQL SERVER IMAGE IN DOCKER

Run the following command in the terminal again to launch the image that was downloaded in Docker:

```
docker run -d --name sql_server_test -e 'ACCEPT_EULA=Y' -e 'SA_PASSWORD=reallyStrongPwd123' -p 1433:1433  
mcr.microsoft.com/mssql/server:2022-latest
```

- **-d**: Launches the docker container in daemon mode, allowing it to run in the background without a terminal window open.
- **--name**: Sets a name for the Docker container. In this example, we are using `sql_server_test`.
- **-e 'ACCEPT_EULA=Y'**: Confirms by agreeing with end user license agreement (EULA) for Docker.
- **-e 'SA_PASSWORD=dockerStrongPwd123'**: Sets the database password. In this example, we are using "dockerStrongPwd123" as the password. In general, I recommend using a strong password
- **-p 1433:1433**: Maps the container to the local port 1433. Default TCP port that SQL Server uses to listen for connections.
- **mcr.microsoft.com/mssql/server:2022-latest**: Selects an image file for Docker to use.

STEP 4: INSTALL THE MS SQL CLI

We need to install sql-cli via npm. Before proceeding, npm should be installed on the Mac. If not, [download npm](#) now.

Run the below command on your terminal.

```
$ npm install -g sql-cli
```

If you get permission error, use the command below:

```
$ sudo npm install -g sql-cli
```

STEP 5: TEST THE INSTALLATION

Once you've installed the sql-cli, test it by logging in using the command below:

```
$ mssql -u sa -p
```

After giving my password, here is the command:

```
$ mssql -u sa - p dockerStrongPwd123
```

Then run `select @@version` to verify the connectivity in the terminal

In the terminal, install dotnet-ef tool by running this command [dotnet tool install --global dotnet-ef]

Migrate the DbContext by running the command [dotnet ef migrations add Initial]

This will create a migration folder in the root folder

Also run the command to push the migration to the database [dotnet ef database update]

Install Azure Data Studio and Connect it to the Database you have created

server: *localhost*

Username: *sa*

Password: *YourPassword*

Run The Application By Typing [dotnet watch run]

LET DIVE INTO THE FRONTEND

Step 1

Open the launchSettings.json file from the Properties folder in the root folder and delete the **"launchUrl": "Swagger"** line so that the application will open the default index.html file.

The launchSettings.json file should look like this

```
{
  "$schema": "https://json.schemastore.org/launchsettings.json",
  "iisSettings": {
    "windowsAuthentication": false,
    "anonymousAuthentication": true,
    "iisExpress": {
      "applicationUrl": "http://localhost:29732",
      "sslPort": 44348
    }
  },
  "profiles": {
    "http": {
      "commandName": "Project",
      "launchBrowser": true,
```

```

"applicationUrl": "http://localhost:5217",
"environmentVariables": {
  "ASPNETCORE_ENVIRONMENT": "Development"
},
"dotnetRunMessages": true
},
"https": {
  "commandName": "Project",
  "launchBrowser": true,
  "applicationUrl": "https://localhost:7273;http://localhost:5217",
  "environmentVariables": {
    "ASPNETCORE_ENVIRONMENT": "Development"
  },
  "dotnetRunMessages": true
},
"IIS Express": {
  "commandName": "IISExpress",
  "launchBrowser": true,
  "environmentVariables": {
    "ASPNETCORE_ENVIRONMENT": "Development"
  }
}
}
}
}
}

```

Create a folder called **wwwroot**, inside this folder, create the css, image and js folders

Outside the wwwroot folder, which is the root of the application, create the index.html

inside the index.html file, paste this code

```

<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="UTF-8">
<meta http-equiv="X-UA-Compatible" content="IE=edge">
<meta name="viewport" content="width=device-width, initial-scale=1">
<title>Projects Manager</title>

<link rel="stylesheet" href="./css/styles.css">
</head>
<body>
<div class="header">
<div class="header-name">HyProject Manager</div>
</div>

<div id="loginContainer" class="sign-in">

```



```

<form action="javascript:void(0);" method="POST" onsubmit="authenticateUser()">
<h1>Sign In</h1>
<div class="empty"></div>
<input type="email" id="username" placeholder="#9993   Username"/>

<input type="password" id="password"
placeholder="#128272   *****"/>
<input type="submit" value="Sign in">
<a onclick="hysignup()">   Not Registered? Sign Up</a>
</form>

</div>
<br/><br/>
<div class="signup" id="signup">
<form action="javascript:void(0);" method="POST" onsubmit="signUP()">
<h1>Sign Up</h1>
<input type="text" name="name" id="name" placeholder="#128100   Full
Name">
<input type="email" name="email" id="email"
placeholder="#9993   user@example.com">
<input type="password" name="password" id="password1"
placeholder="#128274   Password">
<input type="password" name="password" id="password2"
placeholder="#128272   Confirm Password">
<a onclick="hyverify()">   Already Registered? Verify</a>

<input type="submit" value="Sign Up">
</form>

</div>
<div class="veriToken" id="veriToken">
<form action="javascript:void(0);" method="POST" onsubmit="verifyUser()">
<h1>Verify Account</h1>
<input type="text" name="verificationToken " id="token"
placeholder="#128477   Enter the verification token">
<input type="submit" value="Verify">
</form>

</div>

<div id="dashboardContainer" class="hidden">

<div class="sidebar" id="sidebar">
<div class="sideproject" id="hyaddnew">

<form action="javascript:void(0);" method="POST" onsubmit="addProject()"
class="addProject">
<fieldset>
<legend>Add Project</legend>

```

```



```

```

</div>

```

```

<div class="sideproject" id="hyupdate">
<form action="javascript:void(0);" onsubmit="updateProject()">
<fieldset>
<legend>Update Project</legend>


```

```

<div class="sideproject" id="hydelete">

<form action="javascript:void(0);" onsubmit="deleteProject()">
<fieldset>
<legend>Delete Project</legend>


```

```
</div>
```

```
<br/><br/><br/>  
</div>
```

```
<div class="display_projects" id="display_projects">  
<div class="organizer">
```

```
<h2 id="welcomeMessage" class="welcome"></h2>
```

```
<div class="open" id="opener" onclick="opener()">Click to Add, Edit or Delete  
Project</div>
```

```
</div>  
<div class="container" id="container"></div>
```

```
</div>
```

```
</div>
```

```
<script src="js/script.js" asp-append-version="true"></script>  
<script type="text/javascript">  
getProject();  
</script>
```

```
</body>  
</html>
```

Inside the css folder, create the **styles.css** add the following code

```
.header{  
height: 10vh;  
display: flex;
```

```

justify-content: center;
background: rgba(47, 45, 45,1);

border-radius: 16px;
box-shadow: 0 4px 30px rgba(0, 0, 0, 0.1);
backdrop-filter: blur(19.9px);
-webkit-backdrop-filter: blur(19.9px);
border: 1px solid rgba(255, 255, 255, 0.65);

}

.header-name{
font-size: 2rem;
padding-top: 1rem;
padding-bottom: 1rem;
color: rgba(255, 255, 255, 0.9);
font-family: 'Times New Roman', Times, serif;
}

.hidden {
display: none;
}

.card{
width:25rem;
word-wrap: break-word;
/* From https://css.glass */
background: rgba(47, 45, 45,1);
border-radius: 16px;
box-shadow: 0 4px 30px rgba(0, 0, 0, 0.1);
backdrop-filter: blur(13.1px);
-webkit-backdrop-filter: blur(13.1px);
border: 1px solid rgba(255, 255, 255, 0.65);
font-size: 2rem;
font-family: 'Times New Roman', Times, serif;
display: flex;
flex-direction: column;
gap: 1rem;
word-wrap: break-word;

}

.card-1>span{

color:#f06040
}

.card-1{
color: grey;
}

```

```

.container{
display: flex;
flex-direction: row;
gap: 2rem;
flex-wrap: wrap;
}

.card-2{
font-size: 2rem;
font-weight:700;

background: rgba(240, 96, 64, 0.55);
border-radius: 16px;
box-shadow: 0 4px 30px rgba(0, 0, 0, 0.1);
backdrop-filter: blur(15.7px);
-webkit-backdrop-filter: blur(15.7px);
border: 1px solid rgba(240, 96, 64, 1);
justify-content: center;
display: flex;
}

.sign-in{
display: flex;
justify-content: center;
margin-top: 10rem;
}

.sign-in>form{
display: flex;
flex-direction: column;
gap: 1rem;
width: 40rem;
height: 30rem;
background: rgba(255, 255, 255, 0.2);
/* From https://css.glass */
background: rgba(255, 255, 255, 0.74);
border-radius: 16px;
box-shadow: 0 4px 30px rgba(0, 0, 0, 0.1);
backdrop-filter: blur(16.8px);
-webkit-backdrop-filter: blur(16.8px);
border: 1px solid rgba(255, 255, 255, 0.79);

}

.sign-in>form>input{
border-radius: 2rem;

```

```

width: 35rem;
margin-left: 2rem;
margin-right: 2rem;
height: 3rem;
font-size: 2rem;
border-radius: 16px;
/* From https://css.glass */
background: rgba(255, 255, 255, 0.32);
border-radius: 16px;
box-shadow: 0 4px 30px rgba(0, 0, 0, 0.1);
backdrop-filter: blur(3.9px);
-webkit-backdrop-filter: blur(3.9px);
border: 1px solid rgba(255, 255, 255, 0.68);
cursor:pointer;
}
.sign-in>form>a{
border-radius: 2rem;
width: 35rem;
margin-left: 2rem;
margin-right: 2rem;
margin-top: 2rem;
height: 3rem;
font-size: 2rem;
border-radius: 16px;
cursor:pointer;
/* From https://css.glass */
background: rgba(255, 255, 255, 0.32);
border-radius: 16px;
box-shadow: 0 4px 30px rgba(0, 0, 0, 0.1);
backdrop-filter: blur(3.9px);
-webkit-backdrop-filter: blur(3.9px);
border: 1px solid rgba(255, 255, 255, 0.68);

}

```

```

.sign-in>form>h1{
text-align: center;
font-size: 3rem;
margin-top: 2rem;
margin-bottom: 2rem;
}

```

```

body{
background-image: url("../image/2.jpeg");
background-size: cover;
background-repeat: no-repeat;
background-position: center center;

```

```
background-attachment: fixed;
```

```
}
```

```
.signup{  
display: none;  
justify-content: center;  
margin-top: 10rem;  
}
```

```
.signup>form{  
display: flex;  
flex-direction: column;  
gap: 1rem;  
width: 40rem;  
height: 40rem;  
background: rgba(255, 255, 255, 0.2);  
/* From https://css.glass */  
background: rgba(255, 255, 255, 0.74);  
border-radius: 16px;  
box-shadow: 0 4px 30px rgba(0, 0, 0, 0.1);  
backdrop-filter: blur(16.8px);  
-webkit-backdrop-filter: blur(16.8px);  
border: 1px solid rgba(255, 255, 255, 0.79);  
}
```

```
.signup>form>input{  
border-radius: 2rem;  
width: 35rem;  
margin-left: 2rem;  
margin-right: 2rem;  
height: 3rem;  
font-size: 2rem;  
border-radius: 16px;  
/* From https://css.glass */  
background: rgba(255, 255, 255, 0.32);  
border-radius: 16px;  
box-shadow: 0 4px 30px rgba(0, 0, 0, 0.1);  
backdrop-filter: blur(3.9px);  
-webkit-backdrop-filter: blur(3.9px);  
border: 1px solid rgba(255, 255, 255, 0.68);  
cursor:pointer;  
}
```

```
.signup>form>a{  
border-radius: 2rem;  
width: 35rem;  
margin-left: 2rem;  
margin-right: 2rem;
```

```
margin-top: 2rem;
height: 3rem;
font-size: 2rem;
border-radius: 16px;
cursor:pointer;
/* From https://css.glass */
background: rgba(255, 255, 255, 0.32);
border-radius: 16px;
box-shadow: 0 4px 30px rgba(0, 0, 0, 0.1);
backdrop-filter: blur(3.9px);
-webkit-backdrop-filter: blur(3.9px);
border: 1px solid rgba(255, 255, 255, 0.68);

}
```

```
.signup>form>h1{
text-align: center;
font-size: 3rem;
margin-top: 2rem;
margin-bottom: 2rem;
}
```

```
.veriToken{
display: none;
justify-content: center;
margin-top: 10rem;
}
```

```
.veriToken>form{
display: flex;
flex-direction: column;
gap: 1rem;
width: 40rem;
height: 20rem;
background: rgba(255, 255, 255, 0.2);
/* From https://css.glass */
background: rgba(255, 255, 255, 0.74);
border-radius: 16px;
box-shadow: 0 4px 30px rgba(0, 0, 0, 0.1);
backdrop-filter: blur(16.8px);
-webkit-backdrop-filter: blur(16.8px);
border: 1px solid rgba(255, 255, 255, 0.79);
}
```

```
.veriToken>form>input{
border-radius: 2rem;
width: 35rem;
```



```
margin-left: 2rem;
margin-right: 2rem;
height: 3rem;
font-size: 2rem;
border-radius: 16px;
/* From https://css.glass */
background: rgba(255, 255, 255, 0.32);
border-radius: 16px;
box-shadow: 0 4px 30px rgba(0, 0, 0, 0.1);
backdrop-filter: blur(3.9px);
-webkit-backdrop-filter: blur(3.9px);
border: 1px solid rgba(255, 255, 255, 0.68);
cursor:pointer;
}
```

```
.veriToken>form>h1{
text-align: center;
font-size: 3rem;
margin-top: 2rem;
margin-bottom: 2rem;
}
```

```
.sidebar{
display: none;
flex-direction: column;
}
```

```
.sideproject>form>fieldset{
display: flex;
flex-direction: column;
gap:2rem;
width: 40rem;
background: rgba(255, 255, 255, 0.74);
border-radius: 16px;
box-shadow: 0 4px 30px rgba(0, 0, 0, 0.1);
backdrop-filter: blur(16.8px);
-webkit-backdrop-filter: blur(16.8px);
border: 1px solid rgba(255, 255, 255, 0.79);
}
```

```
.sideproject>form>fieldset>input{
border-radius: 2rem;
width: 35rem;
margin-left: 2rem;
margin-right: 2rem;
height: 3rem;
```

```
font-size: 2rem;
border-radius: 16px;
/* From https://css.glass */
background: rgba(255, 255, 255, 0.32);
border-radius: 16px;
box-shadow: 0 4px 30px rgba(0, 0, 0, 0.1);
backdrop-filter: blur(3.9px);
-webkit-backdrop-filter: blur(3.9px);
border: 1px solid rgba(255, 255, 255, 0.68);
cursor:pointer;
}
```

```
.sideproject>form>fieldset>a{
border-radius: 2rem;
width: 35rem;
margin-left: 2rem;
margin-right: 2rem;
height: 3rem;
font-size: 2rem;
border-radius: 16px;
/* From https://css.glass */
background: rgba(255, 255, 255, 0.32);
border-radius: 16px;
box-shadow: 0 4px 30px rgba(0, 0, 0, 0.1);
backdrop-filter: blur(3.9px);
-webkit-backdrop-filter: blur(3.9px);
border: 1px solid rgba(255, 255, 255, 0.68);
cursor:pointer;
}
```

```
.sideproject>form>fieldset>textarea{
font-size: 2rem;
background: rgba(255, 255, 255, 0.32);
border-radius: 16px;
box-shadow: 0 4px 30px rgba(0, 0, 0, 0.1);
backdrop-filter: blur(3.9px);
-webkit-backdrop-filter: blur(3.9px);
border: 1px solid rgba(255, 255, 255, 0.68);
cursor:pointer;
width: 35rem;
margin-left: 2rem;
margin-right: 2rem;
}
.sideproject>form>fieldset>legend{
font-size: 2.5rem;
font-weight: bold;
}
```

```

#hyupdate{
display: none;
}

#hydelete{
display:none;
}


.welcome,.open{
font-size: 2rem;
font-weight:700;
height: 2rem;
background: rgba(240, 96, 64, 0.55);
border-radius: 16px;
box-shadow: 0 4px 30px rgba(0, 0, 0, 0.1);
backdrop-filter: blur(15.7px);
-webkit-backdrop-filter: blur(15.7px);
border: 1px solid rgba(240, 96, 64, 1);
justify-content: center;
display: flex;
}

.sideproject {
position: absolute;
top: 100%;
left: 50%;

transform: translate(-50%, -50%);
}

.organizer{
display: flex;
flex-direction: row;
justify-content: space-evenly;
flex-wrap: wrap;
margin-bottom: 2rem;
}

```

*Inside the js folder, create a **script.js** and add the following code*

```

window.onload = function(){
Projects()
}

const uri = 'api/Projects';
const auth = "api/Auth";
let projects = [];
function getProject(){
fetch(uri)
.then(response => response.json())

```

```

.catch(error=> console.error('Unable to get items.', error))
}
/*
Create a Project
*/
function addProject(){
const addCustomer = document.getElementById("customer");
const addName = document.getElementById("project-name");
const addDescription = document.getElementById("description");
const addDuedate = document.getElementById("duedate");
const addCost = document.getElementById("cost");
const addDeveloper = document.getElementById("developer");
const addStatus = document.getElementById("status");
const projectItem = {
customer: addCustomer.value.trim(),
name: addName.value.trim(),
description: addDescription.value.trim(),
dueDate: addDuedate.value.trim(),
cost : addCost.value.trim(),
developer : addDeveloper.value.trim(),
status: addStatus.value.trim(),
};
fetch(uri, {
method: 'POST',
headers: {
'Accept': 'application/json',
'Content-Type': 'application/json'
},
body: JSON.stringify(projectItem)
})
.then(response => response.json)
.then(()=>{
getProject();
addCustomer.value = "";
addStatus.value = "";
addDescription.value = "";
addDuedate.value = "";
addCost.value = "";
addDeveloper.value = "";
addName.value = "";
})
.catch(error => console.error('Unable to add Project.', error));
}
/*
Update a Project
*/
function updateProject(){
projectId = document.getElementById('edit-id').value.trim();
const projectItem = {
Id : document.getElementById('edit-id').value.trim(),

```

```

customer: document.getElementById('edit-customer').value.trim(),
name: document.getElementById('edit-name').value.trim(),
description: document.getElementById('edit-description').value.trim(),
dueDate: document.getElementById('edit-duedate').value.trim(),
cost: document.getElementById('edit-cost').value.trim(),
developer: document.getElementById('edit-developer').value.trim(),
status: document.getElementById('edit-status').value.trim(),
};
fetch(`${uri}/${projectId}`, {
method: 'PUT',
headers: {
'Accept': 'application/json',
'Content-Type': 'application/json'
},
body: JSON.stringify(projectItem)
})
.then(response => response.json)
.then(() => {
getProject();
document.getElementById('edit-id').value = "";
document.getElementById('edit-customer').value = "";
document.getElementById('edit-name').value = "";
document.getElementById('edit-description').value = "";
document.getElementById('edit-duedate').value = "";
document.getElementById('edit-cost').value = "";
document.getElementById('edit-developer').value = "";
document.getElementById('edit-status').value = "";
})
.catch(error => console.error('Unable to update projects.', error));
return false;
}
/*
DELETE PROJECT
*/
function deleteProject() {
projectId = document.getElementById("delete-id").value.trim();
fetch(`${uri}/${projectId}`, {
method: 'DELETE'
})
.then(() => {
getProject();
document.getElementById("delete-id").value = "";
})
.catch(error => console.error('Unable to delete item.', error));
}
async function Projects(){
try{
const response = await fetch(uri);
const items = await response.json();
const container = document.getElementById("container");

```

```

items.forEach(function(item){
const Card = document.createElement("div");
Card.className = "card"
const customer = document.createElement("div");
customer.className = "card-2";
customer.innerHTML = `&#128188;&nbsp; ${item.customer}`;
Card.appendChild(customer);
const Id = document.createElement("div");
Id.className = "card-1";
Id.innerHTML = `<span >&#128273 ID:</span> ${item.id}`;
Card.appendChild(Id);
const name = document.createElement("div");
name.className = "card-1";
name.innerHTML = `<span >&#128450 Name:</span> ${item.name}`;
Card.appendChild(name);
const description = document.createElement("div");
description.className = "card-1";
description.innerHTML = `<span >&#128452 Description:</span> ${item.description}`;
Card.appendChild(description);
const cost = document.createElement("div");
cost.className = "card-1";
cost.innerHTML = `<span >&#128181 Cost:</span> GHS ${item.cost}`;
Card.appendChild(cost);
const duedate = document.createElement("div");
duedate.className = "card-1";
duedate.innerHTML = `<span >&#128198 Duedate:</span> ${item.dueDate}`;
Card.appendChild(duedate);
const developer = document.createElement("div");
developer.className = "card-1";
developer.innerHTML = `<span >&#128100 Developer:</span> ${item.developer}`;
Card.appendChild(developer);

const status = document.createElement("div");
status.className = "card-1";
status.innerHTML = `<span >&#128640 Status:</span>${item.status}`;
Card.appendChild(status);
container.appendChild(Card)
})
}
catch(error){
console.error("Error fetching Items",error);
}
}
async function verifyUser() {
const token = document.getElementById("token").value.trim();
try {
const response = await fetch(auth+"/verify?token=" + token, {
method: "POST"
});
});

```

```

const data = await response.text();
if (response.ok) {
// Verification successful
alert("Verification successful! " + data);
document.getElementById("signup").style.display = "none";
document.getElementById("veriToken").style.display = "none";
document.getElementById("loginContainer").style.display="flex"
} else {
// Verification failed
alert("Verification failed: " + data);
}
} catch (error) {
console.error("Error verifying user:", error);
}
}
//Signup
async function signUP(){
const name = document.getElementById("name").value;
const email = document.getElementById("email").value;
const password1 = document.getElementById("password1").value;
const password2 = document.getElementById("password2").value;
const register = {
Name: name.trim(),
Email: email.trim(),
Password: password1.trim(),
ConfirmPassword: password2.trim()
}
try{
const response = await fetch(auth+"/register",{
method: "POST",
headers: {
'Accept': 'application/json',
'Content-Type': 'application/json'
},
body: JSON.stringify(register),
})
const data = await response.json();
if (response.ok){
alert(data);
document.getElementById("signup").style.display = "none";
document.getElementById("veriToken").style.display="flex";
}
else{
alert(data)
}
}
catch(error){
console.error("Error registering user:", error);
}
}

```

```

//Login Function
async function authenticateUser() {
const username = document.getElementById("username").value.trim();
const password = document.getElementById("password").value.trim();
const login = {
Email:username,
Password:password
}
try {
const response = await fetch(auth+'/login', {
method: "POST",
headers: {
'Accept': 'application/json',
'Content-Type': 'application/json'
},
body: JSON.stringify(login)
});
const data = await response.json();
if (response.ok) {
// Authentication successful
showDashboard(data);
document.getElementById("loginContainer").style.display = "none";
} else {
// Authentication failed
alert("Authentication failed: " + data);
}
} catch (error) {
console.error("Error authenticating user:", error);
}
}

function showDashboard(data) {
const welcomeMessage = document.getElementById("welcomeMessage");
document.getElementById("signup").style.display = "none";
document.getElementById("veriToken").style.display = "none";
document.getElementById("loginContainer").style.display = "none";
document.getElementById("dashboardContainer").style.display="flex";
;
// Display the username in the welcome message
welcomeMessage.textContent = data;
}

function hysignup(){
document.getElementById("signup").style.display = "flex";
document.getElementById("loginContainer").style.display = "none";
}

function hyverify(){
document.getElementById("veriToken").style.display = "flex";
document.getElementById("signup").style.display = "none";
}

```



```
function opener(){
document.getElementById("sidebar").style.display = "flex";
document.getElementById("display_projects").style.display = "none";
document.getElementById("opener").style.display = 'none';
}

function hyupdate(){
document.getElementById("hyupdate").style.display = "flex"
document.getElementById("hydelete").style.display = "none"
document.getElementById("hyaddnew").style.display = "none"
}

function hydelete(){
document.getElementById("hydelete").style.display = "flex"
document.getElementById("hyaddnew").style.display = "none"
document.getElementById("hyupdate").style.display = "none"
}

function hyaddnew(){
document.getElementById("hyaddnew").style.display = "flex"
document.getElementById("hyupdate").style.display = "none"
document.getElementById("hydelete").style.display = "none"
}

}
```

SECURITY

The script.js file can be visible to the hacker and the hacker can manipulate the code for illegal means but this is how you can beat the system

1. Make a copy of your code and store it internally now the production code, you can change all the variable to a single letter
2. You can copy other rubbish code and add it to your code for it to look bulky and meaningless
3. 500 lines of code can be changed to 18000 lines by adding additional rubbish code
4. Remove all comment in production code and use uglifyjs.net to compress it by removing all spaces in the code
5. Rename all your functions to be a nonsense one
6. We can also minify the html with this website

Example of a dummy code we can add to confuse hackers

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

```

def test():
    """
    A function that performs a series of operations on a list of integers.
    It includes nested loops, conditional statements, and arithmetic operations.
    """
    # Initial setup
    n = 10
    a = 1
    b = 2
    c = 3
    d = 4
    e = 5
    f = 6
    g = 7
    h = 8
    i = 9
    j = 10

    # Main loop
    for i in range(1, n+1):
        # Conditional check
        if i % 2 == 0:
            # Even numbers
            a = a * i
            b = b + i
        else:
            # Odd numbers
            c = c * i
            d = d + i

        # Arithmetic operations
        e = e + a
        f = f + b
        g = g + c
        h = h + d

        # Conditional check
        if i % 3 == 0:
            # Multiples of 3
            i = i * 2
        else:
            # Not multiples of 3
            i = i * 3

        # Arithmetic operations
        j = j + e
        k = k + f
        l = l + g
        m = m + h

        # Conditional check
        if i % 5 == 0:
            # Multiples of 5
            i = i * 4
        else:
            # Not multiples of 5
            i = i * 5

        # Arithmetic operations
        n = n + j
        o = o + k
        p = p + l
        q = q + m

        # Conditional check
        if i % 7 == 0:
            # Multiples of 7
            i = i * 6
        else:
            # Not multiples of 7
            i = i * 7

        # Arithmetic operations
        r = r + n
        s = s + o
        t = t + p
        u = u + q

        # Conditional check
        if i % 11 == 0:
            # Multiples of 11
            i = i * 10
        else:
            # Not multiples of 11
            i = i * 11

        # Arithmetic operations
        v = v + r
        w = w + s
        x = x + t
        y = y + u

        # Conditional check
        if i % 13 == 0:
            # Multiples of 13
            i = i * 12
        else:
            # Not multiples of 13
            i = i * 13

        # Arithmetic operations
        z = z + v
        aa = aa + w
        ab = ab + x
        ac = ac + y

        # Conditional check
        if i % 17 == 0:
            # Multiples of 17
            i = i * 16
        else:
            # Not multiples of 17
            i = i * 17

        # Arithmetic operations
        ad = ad + z
        ae = ae + aa
        af = af + ab
        ag = ag + ac

        # Conditional check
        if i % 19 == 0:
            # Multiples of 19
            i = i * 18
        else:
            # Not multiples of 19
            i = i * 19

        # Arithmetic operations
        ah = ah + ad
        ai = ai + ae
        aj = aj + af
        ak = ak + ag

        # Conditional check
        if i % 23 == 0:
            # Multiples of 23
            i = i * 22
        else:
            # Not multiples of 23
            i = i * 23

        # Arithmetic operations
        al = al + ah
        am = am + ai
        an = an + aj
        ao = ao + ak

        # Conditional check
        if i % 29 == 0:
            # Multiples of 29
            i = i * 28
        else:
            # Not multiples of 29
            i = i * 29

        # Arithmetic operations
        ap = ap + al
        aq = aq + am
        ar = ar + an
        as = as + ao

        # Conditional check
        if i % 31 == 0:
            # Multiples of 31
            i = i * 30
        else:
            # Not multiples of 31
            i = i * 31

        # Arithmetic operations
        at = at + ap
        au = au + aq
        av = av + ar
        aw = aw + as

        # Conditional check
        if i % 37 == 0:
            # Multiples of 37
            i = i * 36
        else:
            # Not multiples of 37
            i = i * 37

        # Arithmetic operations
        ax = ax + at
        ay = ay + au
        az = az + av
        ba = ba + aw

        # Conditional check
        if i % 41 == 0:
            # Multiples of 41
            i = i * 40
        else:
            # Not multiples of 41
            i = i * 41

        # Arithmetic operations
        bb = bb + ax
        bc = bc + ay
        bd = bd + az
        be = be + ba

        # Conditional check
        if i % 43 == 0:
            # Multiples of 43
            i = i * 42
        else:
            # Not multiples of 43
            i = i * 43

        # Arithmetic operations
        bf = bf + bb
        bg = bg + bc
        bh = bh + bd
        bi = bi + be

        # Conditional check
        if i % 47 == 0:
            # Multiples of 47
            i = i * 46
        else:
            # Not multiples of 47
            i = i * 47

        # Arithmetic operations
        bj = bj + bf
        bk = bk + bg
        bl = bl + bh
        bm = bm + bi

        # Conditional check
        if i % 53 == 0:
            # Multiples of 53
            i = i * 52
        else:
            # Not multiples of 53
            i = i * 53

        # Arithmetic operations
        bn = bn + bj
        bo = bo + bk
        bp = bp + bl
        bq = bq + bm

        # Conditional check
        if i % 59 == 0:
            # Multiples of 59
            i = i * 58
        else:
            # Not multiples of 59
            i = i * 59

        # Arithmetic operations
        br = br + bn
        bs = bs + bo
        bt = bt + bp
        bu = bu + bq

        # Conditional check
        if i % 61 == 0:
            # Multiples of 61
            i = i * 60
        else:
            # Not multiples of 61
            i = i * 61

        # Arithmetic operations
        bv = bv + br
        bw = bw + bs
        bx = bx + bt
        by = by + bu

        # Conditional check
        if i % 67 == 0:
            # Multiples of 67
            i = i * 66
        else:
            # Not multiples of 67
            i = i * 67

        # Arithmetic operations
        bz = bz + bv
        ba_2 = ba_2 + bw
        bb_2 = bb_2 + bx
        bc_2 = bc_2 + by

        # Conditional check
        if i % 71 == 0:
            # Multiples of 71
            i = i * 70
        else:
            # Not multiples of 71
            i = i * 71

        # Arithmetic operations
        bd_2 = bd_2 + bz
        be_2 = be_2 + ba_2
        bf_2 = bf_2 + bb_2
        bg_2 = bg_2 + bc_2

        # Conditional check
        if i % 73 == 0:
            # Multiples of 73
            i = i * 72
        else:
            # Not multiples of 73
            i = i * 73

        # Arithmetic operations
        bh_2 = bh_2 + bd_2
        bi_2 = bi_2 + be_2
        bj_2 = bj_2 + bf_2
        bk_2 = bk_2 + bg_2

        # Conditional check
        if i % 79 == 0:
            # Multiples of 79
            i = i * 78
        else:
            # Not multiples of 79
            i = i * 79

        # Arithmetic operations
        bl_2 = bl_2 + bh_2
        bm_2 = bm_2 + bi_2
        bn_2 = bn_2 + bj_2
        bo_2 = bo_2 + bk_2

        # Conditional check
        if i % 83 == 0:
            # Multiples of 83
            i = i * 82
        else:
            # Not multiples of 83
            i = i * 83

        # Arithmetic operations
        bp_2 = bp_2 + bl_2
        bq_2 = bq_2 + bm_2
        br_2 = br_2 + bn_2
        bs_2 = bs_2 + bo_2

        # Conditional check
        if i % 89 == 0:
            # Multiples of 89
            i = i * 88
        else:
            # Not multiples of 89
            i = i * 89

        # Arithmetic operations
        bt_2 = bt_2 + bp_2
        bu_2 = bu_2 + bq_2
        bv_2 = bv_2 + br_2
        bw_2 = bw_2 + bs_2

        # Conditional check
        if i % 97 == 0:
            # Multiples of 97
            i = i * 96
        else:
            # Not multiples of 97
            i = i * 97

        # Arithmetic operations
        bx_2 = bx_2 + bt_2
        by_2 = by_2 + bu_2
        bz_2 = bz_2 + bv_2
        ba_3 = ba_3 + bw_2

        # Conditional check
        if i % 101 == 0:
            # Multiples of 101
            i = i * 100
        else:
            # Not multiples of 101
            i = i * 101

        # Arithmetic operations
        bc_3 = bc_3 + bx_2
        bd_3 = bd_3 + by_2
        be_3 = be_3 + bz_2
        bf_3 = bf_3 + ba_3

        # Conditional check
        if i % 103 == 0:
            # Multiples of 103
            i = i * 102
        else:
            # Not multiples of 103
            i = i * 103

        # Arithmetic operations
        bg_3 = bg_3 + bc_3
        bh_3 = bh_3 + bd_3
        bi_3 = bi_3 + be_3
        bj_3 = bj_3 + bf_3

        # Conditional check
        if i % 107 == 0:
            # Multiples of 107
            i = i * 106
        else:
            # Not multiples of 107
            i = i * 107

        # Arithmetic operations
        bk_3 = bk_3 + bg_3
        bl_3 = bl_3 + bh_3
        bm_3 = bm_3 + bi_3
        bn_3 = bn_3 + bj_3

        # Conditional check
        if i % 113 == 0:
            # Multiples of 113
            i = i * 112
        else:
            # Not multiples of 113
            i = i * 113

        # Arithmetic operations
        bo_3 = bo_3 + bk_3
        bp_3 = bp_3 + bl_3
        bq_3 = bq_3 + bm_3
        br_3 = br_3 + bn_3

        # Conditional check
        if i % 127 == 0:
            # Multiples of 127
            i = i * 126
        else:
            # Not multiples of 127
            i = i * 127

        # Arithmetic operations
        bs_3 = bs_3 + bo_3
        bt_3 = bt_3 + bp_3
        bu_3 = bu_3 + bq_3
        bv_3 = bv_3 + br_3

        # Conditional check
        if i % 131 == 0:
            # Multiples of 131
            i = i * 130
        else:
            # Not multiples of 131
            i = i * 131

        # Arithmetic operations
        bw_3 = bw_3 + bs_3
        bx_3 = bx_3 + bt_3
        by_3 = by_3 + bu_3
        bz_3 = bz_3 + bv_3

        # Conditional check
        if i % 137 == 0:
            # Multiples of 137
            i = i * 136
        else:
            # Not multiples of 137
            i = i * 137

        # Arithmetic operations
        ba_4 = ba_4 + bw_3
        bb_4 = bb_4 + bx_3
        bc_4 = bc_4 + by_3
        bd_4 = bd_4 + bz_3

        # Conditional check
        if i % 149 == 0:
            # Multiples of 149
            i = i * 148
        else:
            # Not multiples of 149
            i = i * 149

        # Arithmetic operations
        be_4 = be_4 + ba_4
        bf_4 = bf_4 + bb_4
        bg_4 = bg_4 + bc_4
        bh_4 = bh_4 + bd_4

        # Conditional check
        if i % 151 == 0:
            # Multiples of 151
            i = i * 150
        else:
            # Not multiples of 151
            i = i * 151

        # Arithmetic operations
        bi_4 = bi_4 + be_4
        bj_4 = bj_4 + bf_4
        bk_4 = bk_4 + bg_4
        bl_4 = bl_4 + bh_4

        # Conditional check
        if i % 157 == 0:
            # Multiples of 157
            i = i * 156
        else:
            # Not multiples of 157
            i = i * 157

        # Arithmetic operations
        bm_4 = bm_4 + bi_4
        bn_4 = bn_4 + bj_4
        bo_4 = bo_4 + bk_4
        bp_4 = bp_4 + bl_4

        # Conditional check
        if i % 163 == 0:
            # Multiples of 163
            i = i * 162
        else:
            # Not multiples of 163
            i = i * 163

        # Arithmetic operations
        bq_4 = bq_4 + bm_4
        br_4 = br_4 + bn_4
        bs_4 = bs_4 + bo_4
        bt_4 = bt_4 + bp_4

        # Conditional check
        if i % 167 == 0:
            # Multiples of 167
            i = i * 166
        else:
            # Not multiples of 167
            i = i * 167

        # Arithmetic operations
        bu_4 = bu_4 + bq_4
        bv_4 = bv_4 + br_4
        bw_4 = bw_4 + bs_4
        bx_4 = bx_4 + bt_4

        # Conditional check
        if i % 173 == 0:
            # Multiples of 173
            i = i * 172
        else:
            # Not multiples of 173
            i = i * 173

        # Arithmetic operations
        by_4 = by_4 + bu_4
        bz_4 = bz_4 + bv_4
        ba_5 = ba_5 + bw_4
        bb_5 = bb_5 + bx_4

        # Conditional check
        if i % 179 == 0:
            # Multiples
```

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

```

1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206
207
208
209
210
211
212
213
214
215
216
217
218
219
220
221
222
223
224
225
226
227
228
229
230
231
232
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
260
261
262
263
264
265
266
267
268
269
270
271
272
273
274
275
276
277
278
279
280
281
282
283
284
285
286
287
288
289
290
291
292
293
294
295
296
297
298
299
300
301
302
303
304
305
306
307
308
309
310
311
312
313
314
315
316
317
318
319
320
321
322
323
324
325
326
327
328
329
330
331
332
333
334
335
336
337
338
339
340
341
342
343
344
345
346
347
348
349
350
351
352
353
354
355
356
357
358
359
360
361
362
363
364
365
366
367
368
369
370
371
372
373
374
375
376
377
378
379
380
381
382
383
384
385
386
387
388
389
390
391
392
393
394
395
396
397
398
399
400
401
402
403
404
405
406
407
408
409
410
411
412
413
414
415
416
417
418
419
420
421
422
423
424
425
426
427
428
429
430
431
432
433
434
435
436
437
438
439
440
441
442
443
444
445
446
447
448
449
450
451
452
453
454
455
456
457
458
459
460
461
462
463
464
465
466
467
468
469
470
471
472
473
474
475
476
477
478
479
480
481
482
483
484
485
486
487
488
489
490
491
492
493
494
495
496
497
498
499
500
501
502
503
504
505
506
507
508
509
510
511
512
513
514
515
516
517
518
519
520
521
522
523
524
525
526
527
528
529
530
531
532
533
534
535
536
537
538
539
540
541
542
543
544
545
546
547
548
549
550
551
552
553
554
555
556
557
558
559
560
561
562
563
564
565
566
567
568
569
570
571
572
573
574
575
576
577
578
579
580
581
582
583
584
585
586
587
588
589
590
591
592
593
594
595
596
597
598
599
600
601
602
603
604
605
606
607
608
609
610
611
612
613
614
615
616
617
618
619
620
621
622
623
624
625
626
627
628
629
630
631
632
633
634
635
636
637
638
639
640
641
642
643
644
645
646
647
648
649
650
651
652
653
654
655
656
657
658
659
660
661
662
663
664
665
666
667
668
669
670
671
672
673
674
675
676
677
678
679
680
681
682
683
684
685
686
687
688
689
690
691
692
693
694
695
696
697
698
699
700
701
702
703
704
705
706
707
708
709
710
711
712
713
714
715
716
717
718
719
720
721
722
723
724
725
726
727
728
729
730
731
732
733
734
735
736
737
738
739
740
741
742
743
744
745
746
747
748
749
750
751
752
753
754
755
756
757
758
759
760
761
762
763
764
765
766
767
768
769
770
771
772
773
774
775
776
777
778
779
780
781
782
783
784
785
786
787
788
789
790
791
792
793
794
795
796
797
798
799
800
801
802
803
804
805
806
807
808
809
810
811
812
813
814
815
816
817
818
819
820
821
822
823
824
825
826
827
828
829
830
831
832
833
834
835
836
837
838
839
840
841
842
843
844
845
846
847
848
849
850
851
852
853
854
855
856
857
858
859
860
861
862
863
864
865
866
867
868
869
870
871
872
873
874
875
876
877
878
879
880
881
882
883
884
885
886
887
888
889
890
891
892
893
894
895
896
897
898
899
900
901
902
903
904
905
906
907
908
909
910
911
912
913
914
915
916
917
918
919
920
921
922
923
924
925
926
927
928
929
930
931
932
933
934
935
936
937
938
939
940
941
942
943
944
945
946
947
948
949
950
951
952
953
954
955
956
957
958
959
960
961
962
963
964
965
966
967
968
969
970
971
972
973
974
975
976
977
978
979
980
981
982
983
984
985
986
987
988
989
990
991
992
993
994
995
996
997
998
999
1000
1001
1002
1003
1004
1005
1006
1007
1008
1009
1010
1011
1012
1013
1014
1015
1016
1017
1018
1019
1020
1021
1022
1023
1024
1025
1026
1027
1028
1029
1030
1031
1032
1033
1034
1035
1036
1037
1038
1039
1040

```

[illegible]

[illegible]


```

1  1  2  3  4  5  6  7  8  9  10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80 81 82 83 84 85 86 87 88 89 90 91 92 93 94 95 96 97 98 99 100 101 102 103 104 105 106 107 108 109 110 111 112 113 114 115 116 117 118 119 120 121 122 123 124 125 126 127 128 129 130 131 132 133 134 135 136 137 138 139 140 141 142 143 144 145 146 147 148 149 150 151 152 153 154 155 156 157 158 159 160 161 162 163 164 165 166 167 168 169 170 171 172 173 174 175 176 177 178 179 180 181 182 183 184 185 186 187 188 189 190 191 192 193 194 195 196 197 198 199 200 201 202 203 204 205 206 207 208 209 210 211 212 213 214 215 216 217 218 219 220 221 222 223 224 225 226 227 228 229 230 231 232 233 234 235 236 237 238 239 240 241 242 243 244 245 246 247 248 249 250 251 252 253 254 255 256 257 258 259 260 261 262 263 264 265 266 267 268 269 270 271 272 273 274 275 276 277 278 279 280 281 282 283 284 285 286 287 288 289 290 291 292 293 294 295 296 297 298 299 300 301 302 303 304 305 306 307 308 309 310 311 312 313 314 315 316 317 318 319 320 321 322 323 324 325 326 327 328 329 330 331 332 333 334 335 336 337 338 339 340 341 342 343 344 345 346 347 348 349 350 351 352 353 354 355 356 357 358 359 360 361 362 363 364 365 366 367 368 369 370 371 372 373 374 375 376 377 378 379 380 381 382 383 384 385 386 387 388 389 390 391 392 393 394 395 396 397 398 399 400 401 402 403 404 405 406 407 408 409 410 411 412 413 414 415 416 417 418 419 420 421 422 423 424 425 426 427 428 429 430 431 432 433 434 435 436 437 438 439 440 441 442 443 444 445 446 447 448 449 450 451 452 453 454 455 456 457 458 459 460 461 462 463 464 465 466 467 468 469 470 471 472 473 474 475 476 477 478 479 480 481 482 483 484 485 486 487 488 489 490 491 492 493 494 495 496 497 498 499 500 501 502 503 504 505 506 507 508 509 510 511 512 513 514 515 516 517 518 519 520 521 522 523 524 525 526 527 528 529 530 531 532 533 534 535 536 537 538 539 540 541 542 543 544 545 546 547 548 549 550 551 552 553 554 555 556 557 558 559 560 561 562 563 564 565 566 567 568 569 570 571 572 573 574 575 576 577 578 579 580 581 582 583 584 585 586 587 588 589 590 591 592 593 594 595 596 597 598 599 600 601 602 603 604 605 606 607 608 609 610 611 612 613 614 615 616 617 618 619 620 621 622 623 624 625 626 627 628 629 630 631 632 633 634 635 636 637 638 639 640 641 642 643 644 645 646 647 648 649 650 651 652 653 654 655 656 657 658 659 660 661 662 663 664 665 666 667 668 669 670 671 672 673 674 675 676 677 678 679 680 681 682 683 684 685 686 687 688 689 690 691 692 693 694 695 696 697 698 699 700 701 702 703 704 705 706 707 708 709 710 711 712 713 714 715 716 717 718 719 720 721 722 723 724 725 726 727 728 729 730 731 732 733 734 735 736 737 738 739 740 741 742 743 744 745 746 747 748 749 750 751 752 753 754 755 756 757 758 759 760 761 762 763 764 765 766 767 768 769 770 771 772 773 774 775 776 777 778 779 780 781 782 783 784 785 786 787 788 789 790 791 792 793 794 795 796 797 798 799 800 801 802 803 804 805 806 807 808 809 810 811 812 813 814 815 816 817 818 819 820 821 822 823 824 825 826 827 828 829 830 831 832 833 834 835 836 837 838 839 840 841 842 843 844 845 846 847 848 849 850 851 852 853 854 855 856 857 858 859 860 861 862 863 864 865 866 867 868 869 870 871 872 873 874 875 876 877 878 879 880 881 882 883 884 885 886 887 888 889 890 891 892 893 894 895 896 897 898 899 900 901 902 903 904 905 906 907 908 909 910 911 912 913 914 915 916 917 918 919 920 921 922 923 924 925 926 927 928 929 930 931 932 933 934 935 936 937 938 939 940 941 942 943 944 945 946 947 948 949 950 951 952 953 954 955 956 957 958 959 960 961 962 963 964 965 966 967 968 969 970 971 972 973 974 975 976 977 978 979 980 981 982 983 984 985 986 987 988 989 990 991 992 993 994 995 996 997 998 999 1000

```

[illegible]

[illegible]

[illegible]


```

def a = 0; b = 1; c = 2; d = 3; e = 4; f = 5; g = 6; h = 7; i = 8; j = 9; k = 10; l = 11; m = 12; n = 13; o = 14; p = 15; q = 16; r = 17; s = 18; t = 19; u = 20; v = 21; w = 22; x = 23; y = 24; z = 25; A = 26; B = 27; C = 28; D = 29; E = 30; F = 31; G = 32; H = 33; I = 34; J = 35; K = 36; L = 37; M = 38; N = 39; O = 40; P = 41; Q = 42; R = 43; S = 44; T = 45; U = 46; V = 47; W = 48; X = 49; Y = 50; Z = 51; a1 = 52; a2 = 53; a3 = 54; a4 = 55; a5 = 56; a6 = 57; a7 = 58; a8 = 59; a9 = 60; a10 = 61; a11 = 62; a12 = 63; a13 = 64; a14 = 65; a15 = 66; a16 = 67; a17 = 68; a18 = 69; a19 = 70; a20 = 71; a21 = 72; a22 = 73; a23 = 74; a24 = 75; a25 = 76; a26 = 77; a27 = 78; a28 = 79; a29 = 80; a30 = 81; a31 = 82; a32 = 83; a33 = 84; a34 = 85; a35 = 86; a36 = 87; a37 = 88; a38 = 89; a39 = 90; a40 = 91; a41 = 92; a42 = 93; a43 = 94; a44 = 95; a45 = 96; a46 = 97; a47 = 98; a48 = 99; a49 = 100; a50 = 101; a51 = 102; a52 = 103; a53 = 104; a54 = 105; a55 = 106; a56 = 107; a57 = 108; a58 = 109; a59 = 110; a60 = 111; a61 = 112; a62 = 113; a63 = 114; a64 = 115; a65 = 116; a66 = 117; a67 = 118; a68 = 119; a69 = 120; a70 = 121; a71 = 122; a72 = 123; a73 = 124; a74 = 125; a75 = 126; a76 = 127; a77 = 128; a78 = 129; a79 = 130; a80 = 131; a81 = 132; a82 = 133; a83 = 134; a84 = 135; a85 = 136; a86 = 137; a87 = 138; a88 = 139; a89 = 140; a90 = 141; a91 = 142; a92 = 143; a93 = 144; a94 = 145; a95 = 146; a96 = 147; a97 = 148; a98 = 149; a99 = 150; a100 = 151; a101 = 152; a102 = 153; a103 = 154; a104 = 155; a105 = 156; a106 = 157; a107 = 158; a108 = 159; a109 = 160; a110 = 161; a111 = 162; a112 = 163; a113 = 164; a114 = 165; a115 = 166; a116 = 167; a117 = 168; a118 = 169; a119 = 170; a120 = 171; a121 = 172; a122 = 173; a123 = 174; a124 = 175; a125 = 176; a126 = 177; a127 = 178; a128 = 179; a129 = 180; a130 = 181; a131 = 182; a132 = 183; a133 = 184; a134 = 185; a135 = 186; a136 = 187; a137 = 188; a138 = 189; a139 = 190; a140 = 191; a141 = 192; a142 = 193; a143 = 194; a144 = 195; a145 = 196; a146 = 197; a147 = 198; a148 = 199; a149 = 200; a150 = 201; a151 = 202; a152 = 203; a153 = 204; a154 = 205; a155 = 206; a156 = 207; a157 = 208; a158 = 209; a159 = 210; a160 = 211; a161 = 212; a162 = 213; a163 = 214; a164 = 215; a165 = 216; a166 = 217; a167 = 218; a168 = 219; a169 = 220; a170 = 221; a171 = 222; a172 = 223; a173 = 224; a174 = 225; a175 = 226; a176 = 227; a177 = 228; a178 = 229; a179 = 230; a180 = 231; a181 = 232; a182 = 233; a183 = 234; a184 = 235; a185 = 236; a186 = 237; a187 = 238; a188 = 239; a189 = 240; a190 = 241; a191 = 242; a192 = 243; a193 = 244; a194 = 245; a195 = 246; a196 = 247; a197 = 248; a198 = 249; a199 = 250; a200 = 251; a201 = 252; a202 = 253; a203 = 254; a204 = 255; a205 = 256; a206 = 257; a207 = 258; a208 = 259; a209 = 260; a210 = 261; a211 = 262; a212 = 263; a213 = 264; a214 = 265; a215 = 266; a216 = 267; a217 = 268; a218 = 269; a219 = 270; a220 = 271; a221 = 272; a222 = 273; a223 = 274; a224 = 275; a225 = 276; a226 = 277; a227 = 278; a228 = 279; a229 = 280; a230 = 281; a231 = 282; a232 = 283; a233 = 284; a234 = 285; a235 = 286; a236 = 287; a237 = 288; a238 = 289; a239 = 290; a240 = 291; a241 = 292; a242 = 293; a243 = 294; a244 = 295; a245 = 296; a246 = 297; a247 = 298; a248 = 299; a249 = 300; a250 = 301; a251 = 302; a252 = 303; a253 = 304; a254 = 305; a255 = 306; a256 = 307; a257 = 308; a258 = 309; a259 = 310; a260 = 311; a261 = 312; a262 = 313; a263 = 314; a264 = 315; a265 = 316; a266 = 317; a267 = 318; a268 = 319; a269 = 320; a270 = 321; a271 = 322; a272 = 323; a273 = 324; a274 = 325; a275 = 326; a276 = 327; a277 = 328; a278 = 329; a279 = 330; a280 = 331; a281 = 332; a282 = 333; a283 = 334; a284 = 335; a285 = 336; a286 = 337; a287 = 338; a288 = 339; a289 = 340; a290 = 341; a291 = 342; a292 = 343; a293 = 344; a294 = 345; a295 = 346; a296 = 347; a297 = 348; a298 = 349; a299 = 350; a300 = 351; a301 = 352; a302 = 353; a303 = 354; a304 = 355; a305 = 356; a306 = 357; a307 = 358; a308 = 359; a309 = 360; a310 = 361; a311 = 362; a312 = 363; a313 = 364; a314 = 365; a315 = 366; a316 = 367; a317 = 368; a318 = 369; a319 = 370; a320 = 371; a321 = 372; a322 = 373; a323 = 374; a324 = 375; a325 = 376; a326 = 377; a327 = 378; a328 = 379; a329 = 380; a330 = 381; a331 = 382; a332 = 383; a333 = 384; a334 = 385; a335 = 386; a336 = 387; a337 = 388; a338 = 389; a339 = 390; a340 = 391; a341 = 392; a342 = 393; a343 = 394; a344 = 395; a345 = 396; a346 = 397; a347 = 398; a348 = 399; a349 = 400; a350 = 401; a351 = 402; a352 = 403; a353 = 404; a354 = 405; a355 = 406; a356 = 407; a357 = 408; a358 = 409; a359 = 410; a360 = 411; a361 = 412; a362 = 413; a363 = 414; a364 = 415; a365 = 416; a366 = 417; a367 = 418; a368 = 419; a369 = 420; a370 = 421; a371 = 422; a372 = 423; a373 = 424; a374 = 425; a375 = 426; a376 = 427; a377 = 428; a378 = 429; a379 = 430; a380 = 431; a381 = 432; a382 = 433; a383 = 434; a384 = 435; a385 = 436; a386 = 437; a387 = 438; a388 = 439; a389 = 440; a390 = 441; a391 = 442; a392 = 443; a393 = 444; a394 = 445; a395 = 
```


[illegible]

[illegible]

```

1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206
207
208
209
210
211
212
213
214
215
216
217
218
219
220
221
222
223
224
225
226
227
228
229
230
231
232
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
260
261
262
263
264
265
266
267
268
269
270
271
272
273
274
275
276
277
278
279
280
281
282
283
284
285
286
287
288
289
290
291
292
293
294
295
296
297
298
299
300
301
302
303
304
305
306
307
308
309
310
311
312
313
314
315
316
317
318
319
320
321
322
323
324
325
326
327
328
329
330
331
332
333
334
335
336
337
338
339
340
341
342
343
344
345
346
347
348
349
350
351
352
353
354
355
356
357
358
359
360
361
362
363
364
365
366
367
368
369
370
371
372
373
374
375
376
377
378
379
380
381
382
383
384
385
386
387
388
389
390
391
392
393
394
395
396
397
398
399
400
401
402
403
404
405
406
407
408
409
410
411
412
413
414
415
416
417
418
419
420
421
422
423
424
425
426
427
428
429
430
431
432
433
434
435
436
437
438
439
440
441
442
443
444
445
446
447
448
449
450
451
452
453
454
455
456
457
458
459
460
461
462
463
464
465
466
467
468
469
470
471
472
473
474
475
476
477
478
479
480
481
482
483
484
485
486
487
488
489
490
491
492
493
494
495
496
497
498
499
500
501
502
503
504
505
506
507
508
509
510
511
512
513
514
515
516
517
518
519
520
521
522
523
524
525
526
527
528
529
530
531
532
533
534
535
536
537
538
539
540
541
542
543
544
545
546
547
548
549
550
551
552
553
554
555
556
557
558
559
560
561
562
563
564
565
566
567
568
569
570
571
572
573
574
575
576
577
578
579
580
581
582
583
584
585
586
587
588
589
590
591
592
593
594
595
596
597
598
599
600
601
602
603
604
605
606
607
608
609
610
611
612
613
614
615
616
617
618
619
620
621
622
623
624
625
626
627
628
629
630
631
632
633
634
635
636
637
638
639
640
641
642
643
644
645
646
647
648
649
650
651
652
653
654
655
656
657
658
659
660
661
662
663
664
665
666
667
668
669
670
671
672
673
674
675
676
677
678
679
680
681
682
683
684
685
686
687
688
689
690
691
692
693
694
695
696
697
698
699
700
701
702
703
704
705
706
707
708
709
710
711
712
713
714
715
716
717
718
719
720
721
722
723
724
725
726
727
728
729
730
731
732
733
734
735
736
737
738
739
740
741
742
743
744
745
746
747
748
749
750
751
752
753
754
755
756
757
758
759
760
761
762
763
764
765
766
767
768
769
770
771
772
773
774
775
776
777
778
779
780
781
782
783
784
785
786
787
788
789
790
791
792
793
794
795
796
797
798
799
800
801
802
803
804
805
806
807
808
809
810
811
812
813
814
815
816
817
818
819
820
821
822
823
824
825
826
827
828
829
830
831
832
833
834
835
836
837
838
839
840
841
842
843
844
845
846
847
848
849
850
851
852
853
854
855
856
857
858
859
860
861
862
863
864
865
866
867
868
869
870
871
872
873
874
875
876
877
878
879
880
881
882
883
884
885
886
887
888
889
890
891
892
893
894
895
896
897
898
899
900
901
902
903
904
905
906
907
908
909
910
911
912
913
914
915
916
917
918
919
920
921
922
923
924
925
926
927
928
929
930
931
932
933
934
935
936
937
938
939
940
941
942
943
944
945
946
947
948
949
950
951
952
953
954
955
956
957
958
959
960
961
962
963
964
965
966
967
968
969
970
971
972
973
974
975
976
977
978
979
980
981
982
983
984
985
986
987
988
989
990
991
992
993
994
995
996
997
998
999
1000
1001
1002
1003
1004
1005
1006
1007
1008
1009
1010
1011
1012
1013
1014
1015
1016
1017
1018
1019
1020
1021
1022
1023
1024
1025
1026
1027
1028
1029
1030
1031
1032
1033
1034
1035
1036
1037
1038
1039
1040

```

```

1  "data-property": "true"
2  [
3  ]
4  [
5  ]
6  [
7  ]
8  [
9  ]
10 [
11 ]
12 [
13 ]
14 [
15 ]
16 [
17 ]
18 [
19 ]
20 [
21 ]
22 [
23 ]
24 [
25 ]
26 [
27 ]
28 [
29 ]
30 [
31 ]
32 [
33 ]
34 [
35 ]
36 [
37 ]
38 [
39 ]
40 [
41 ]
42 [
43 ]
44 [
45 ]
46 [
47 ]
48 [
49 ]
50 [
51 ]
52 [
53 ]
54 [
55 ]
56 [
57 ]
58 [
59 ]
60 [
61 ]
62 [
63 ]
64 [
65 ]
66 [
67 ]
68 [
69 ]
70 [
71 ]
72 [
73 ]
74 [
75 ]
76 [
77 ]
78 [
79 ]
80 [
81 ]
82 [
83 ]
84 [
85 ]
86 [
87 ]
88 [
89 ]
90 [
91 ]
92 [
93 ]
94 [
95 ]
96 [
97 ]
98 [
99 ]
100 [
101 ]
102 [
103 ]
104 [
105 ]
106 [
107 ]
108 [
109 ]
110 [
111 ]
112 [
113 ]
114 [
115 ]
116 [
117 ]
118 [
119 ]
120 [
121 ]
122 [
123 ]
124 [
125 ]
126 [
127 ]
128 [
129 ]
130 [
131 ]
132 [
133 ]
134 [
135 ]
136 [
137 ]
138 [
139 ]
140 [
141 ]
142 [
143 ]
144 [
145 ]
146 [
147 ]
148 [
149 ]
150 [
151 ]
152 [
153 ]
154 [
155 ]
156 [
157 ]
158 [
159 ]
160 [
161 ]
162 [
163 ]
164 [
165 ]
166 [
167 ]
168 [
169 ]
170 [
171 ]
172 [
173 ]
174 [
175 ]
176 [
177 ]
178 [
179 ]
180 [
181 ]
182 [
183 ]
184 [
185 ]
186 [
187 ]
188 [
189 ]
190 [
191 ]
192 [
193 ]
194 [
195 ]
196 [
197 ]
198 [
199 ]
200 [
201 ]
202 [
203 ]
204 [
205 ]
206 [
207 ]
208 [
209 ]
210 [
211 ]
212 [
213 ]
214 [
215 ]
216 [
217 ]
218 [
219 ]
220 [
221 ]
222 [
223 ]
224 [
225 ]
226 [
227 ]
228 [
229 ]
230 [
231 ]
232 [
233 ]
234 [
235 ]
236 [
237 ]
238 [
239 ]
240 [
241 ]
242 [
243 ]
244 [
245 ]
246 [
247 ]
248 [
249 ]
250 [
251 ]
252 [
253 ]
254 [
255 ]
256 [
257 ]
258 [
259 ]
260 [
261 ]
262 [
263 ]
264 [
265 ]
266 [
267 ]
268 [
269 ]
270 [
271 ]
272 [
273 ]
274 [
275 ]
276 [
277 ]
278 [
279 ]
280 [
281 ]
282 [
283 ]
284 [
285 ]
286 [
287 ]
288 [
289 ]
290 [
291 ]
292 [
293 ]
294 [
295 ]
296 [
297 ]
298 [
299 ]
300 [
301 ]
302 [
303 ]
304 [
305 ]
306 [
307 ]
308 [
309 ]
310 [
311 ]
312 [
313 ]
314 [
315 ]
316 [
317 ]
318 [
319 ]
320 [
321 ]
322 [
323 ]
324 [
325 ]
326 [
327 ]
328 [
329 ]
330 [
331 ]
332 [
333 ]
334 [
335 ]
336 [
337 ]
338 [
339 ]
340 [
341 ]
342 [
343 ]
344 [
345 ]
346 [
347 ]
348 [
349 ]
350 [
351 ]
352 [
353 ]
354 [
355 ]
356 [
357 ]
358 [
359 ]
360 [
361 ]
362 [
363 ]
364 [
365 ]
366 [
367 ]
368 [
369 ]
370 [
371 ]
372 [
373 ]
374 [
375 ]
376 [
377 ]
378 [
379 ]
380 [
381 ]
382 [
383 ]
384 [
385 ]
386 [
387 ]
388 [
389 ]
390 [
391 ]
392 [
393 ]
394 [
395 ]
396 [
397 ]
398 [
399 ]
399 ]

```

[illegible]