There are several types of common concrete cracks namely hairline cracks which usually develop in concrete foundation as the concrete cures, shrinkage cracks which occur while the concrete is curing, settlement cracks which happen when part of concrete sinks or when the ground underneath the slab isn't compacted properly as well as structural cracks which form due to incorrect design.

Concrete cracks may endanger the safety and durability of a building if not eing identified quickly and left untreated. Hence, your job as an AI engineer is tasked to perform image classification to classify concretes with or without cracks. Your developed model is impactful and may save thousands of lives.

The criteria of the project are as follows:

1. Link to the dataset: https://data.mendeley.com/datasets/5y9wdsg2zt/2
2. Carry out the whole machine learning workflow for this project (Problem Formulation →Data Preparation→Model Development→Model Deployment)
3. For criteria, try to achieve training and validation accuracy of more than 90%
4. Make sure your model is not overfitting.
5. You may apply transfer learning.
6. When uploading to GitHub, make sure your whole project is presentable:
    a. Make sure you have a complete README file.
    b. You do not need to upload the data onto GitHub, just mention the data source in your README.
    c. Make sure you write your code following the good practices such as following variable naming conventions and writing concise comments to improve the readability of your code.
7. For model deployment, try to use the model to make some predictions with your test data.

Files to be submitted and uploaded to GitHub and LMS (submission link will be given on the assessment day):

1) Training, deployment scripts and classes (GitHub and LMS)
2) Saved model in .h5 format and scalers (if any) in .pkl file format. (GitHub and LMS)
3) Training process plotted using Tensorboard can be snipped and saved as image file format (LMS) and use EarlyStopping callback to prevent overfitting.

4) The architecture of the model should be plotted using plot_model function and saved as .png file format. Include the image in README.md and also upload to LMS. (GitHub and LMS)
5) Performance of the model and the reports can be snipped and saved as image file to be included in the zip folder for LMS submission. (LMS and GitHub)
6) Include your GitHub URL directing to your assessment 2 in a text file then submit to LMS. (LMS)
7) **Don't forget to credit/cite the source of the data on your GitHub page** https://data.mendeley.com/datasets/5y9wdsg2zt/2

*Please zip all the required files into one folder then submit to LMS.

**Please save model in a different folder to GitHub.

***You **DO NOT** need to upload the dataset to your GitHub or your zip file, just make sure you cite the data source properly in your **README.md** (the dataset size is large)

Complete the assessment and submit the files to LMS and GitHub by 5pm. Good Luck!!!

TIPS: For Google Colab user, you may use the following the code to download the dataset inside Google Colab Environment without uploading to Google Colab. Just type the following code in the code cell and execute.

Step 1) To download the dataset:

```
!wget https://prod-dcd-datasets-cache-zipfiles.s3.eu-west-1.amazonaws.com/5y9wdsg2zt-2.zip
```

Step 2) To unzip and extract into a new folder:

```
!unzip "5y9wdsg2zt-2.zip"
!unrar x "Concrete Crack Images for Classification.rar" "dataset"
```

Step 3) All images should now appear in the folder named 'dataset'.

	100%	50%	0%
Task Completion (30%)	Scripts can be executed without any error on trainer's local machine.	-	Scripts fail to be executed on trainer's local machine.
Project requirements (30%)	Able to achieve the objectives of the project using relevant and appropriate approach.	Able to achieve the objectives of the project but using inappropriate approach such as brute forcing the solution.	Fail to achieve the objectives of the project.
Image preprocessing (30%)	Demonstrates strong understanding on the objectives of the project and performs relevant approach to process the data. Necessary image processing techniques such as, brightness correction, image filtering or image resaturation are performed and well justified.	Shows comprehensive understanding of the objectives of the project but uses incorrect or irrelevant approach to process the image.	Shows limited understanding of the objectives of the project. Absence of image processing section in the code.
Code readability (5%)	Involves the usage of functions or methods for repeated tasks. Codes are easily readable and justified by including comments and description texts.	Minimal usage of functions or methods for repeated tasks. Available comments and descriptions but lack of details.	No usage of functions or methods for repeated tasks. Codes are difficult to read and understand. Missing descriptions and comments.
GitHub repo (4%)	Detailed and clear instructions of the project on README.md. Results such as graphs are also included in README.md as part of the project description.	Project successfully uploaded to GitHub repo but with incomplete README.md. Missing descriptions, instructions, and results.	Fails to upload project to GitHub repo and missing README.md
PEP8 compliance (1%)	Fully complies with PEP 8 Standard	Partially complies with PEP 8 Standard	Fails to comply with PEP 8 Standard
Total (100%)			