

Preparation for TensorFlow Developer Certification

Your Name

Title

Company

About Me

Your Image

Your Name

Latest Work Experiences:

- Your Role, X Company
- Your Role, X Company

Year - present
Year - Year

Education:

- Y University
Master of XX
- X University
Bachelor of XX

Year - Year

Year - Year

Ground Rules

Observe the following rules to ensure a supportive, inclusive, and engaging classes



Give full attention
in class



Mute your microphone
when you're not talking



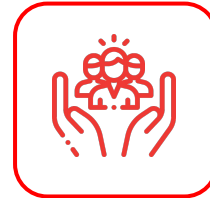
Keep your
camera on



Turn on the CC Feature
on Meet



Use raise hand or chat
to ask questions



Make this room a safe place
to learn and share

Outline **Session**

- Introduction to **TensorFlow Developer Certification**
- Exam **Simulation**
- Exam **Tips**



Introduction to TensorFlow Developer Certification

TensorFlow Developer Certification

TensorFlow Developer Certification (TFDC) demonstrates your proficiency in **using TensorFlow to solve deep learning and ML problems**



About TFDC Exam

- Length: Five hours
- Registration fee: \$100
- Languages: English
- Exam format:
 - This exam is an online, performance-based test that requires implementing TensorFlow
 - In order to take the exam, you will install the TensorFlow Exam plugin using the PyCharm

Registration Process

- Please prepare your **primary ID** (non-expired and including photograph, full name, and signature)
- To register for the exam, go [here](#)
- You have **6 months** from the date of purchasing the exam to take the exam before your purchase expires
- After you download the plugin and sign in, your exam will be provided to you
- Please note that you must **register** and **pay** for each exam attempt

Skills Checklist

The TFD Certification exam will test you on this set of skills:

- TensorFlow developer skills
- Building and training neural network models using TensorFlow 2.x
- Image classification
- Natural language processing (NLP)
- Time series, sequences, and predictions

Setting Environment

- Install Python 3.8
- Install PyCharm 2021.3
- Check your GPU driver
- Create a Test Virtual Environment that uses TensorFlow in PyCharm
 - Create a new PyCharm project
 - Install TensorFlow and related packages

```
tensorflow==2.7.0  
  
tensorflow-datasets==4.4.0  
Pillow==8.4.0  
pandas==1.3.4  
numpy==1.21.4  
scipy==1.7.3
```

TensorFlow Developer Certification Benefits

- Learn something new
- Showcase your skills
- Receive community recognition
- Find TensorFlow talent

TensorFlow Certificate Network

Find TensorFlow Developers who have passed the certification exam to help you with your machine learning and deep learning tasks. This level one certificate exam tests a developer's foundational knowledge of integrating machine learning into tools and applications. The certificate program requires an understanding of building TensorFlow models using Computer Vision, Convolutional Neural Networks, Natural Language Processing, and real-world image data and strategies. You can learn more about this certificate program on our website.

The screenshot displays the TensorFlow Certificate Network interface. At the top, there is a search bar with 'Indonesia' entered, and filters for 'Certification Type', 'Experience', and 'Region'. Below the search bar is a world map with pins indicating certified developers in Indonesia. Below the map, there is a grid of four developer profiles, each featuring a profile picture, name, location, experience, and certification details.

Developer	Location	Experience	Certification Type	Grant date	Expiration date
Harvind Philippe	Indonesia	2-5 years	TensorFlow Developer	4/15/2020	4/15/2022
Aji Setyoko	Indonesia	2-5 years	TensorFlow Developer	5/6/2020	5/6/2022
Andhya Aulianto	Indonesia	> 5 years	TensorFlow Developer	6/13/2020	6/13/2022
Waimond Handoyo	Indonesia	< 1 year	TensorFlow Developer	6/23/2020	6/23/2022

What They Say about **TFDC**



Developing AI Products
for Coral Reefs with
Indonesia Research Agency

Bagus Satya Mas
| Bangkit 2021 Graduate, Certified TFD
| Data Scientist at BISA AI



Exam Simulation

TFDC Exam Simulation

This will provide **three different packages** that simulate what you will be facing when taking the actual TDC exam from the Google Developers Program



Simulation Packages

- Simulation **A**
 - Consist of 5 Machine Learning Problems
- Simulation **B**
 - Consist of 5 machine learning problems.
 - The datasets used in Simulation B are different from the dataset in Simulation B.
- Simulation **C**
 - Consist of 5 machine learning problems.
 - The datasets used in Simulation B are different from the dataset in Simulation A.

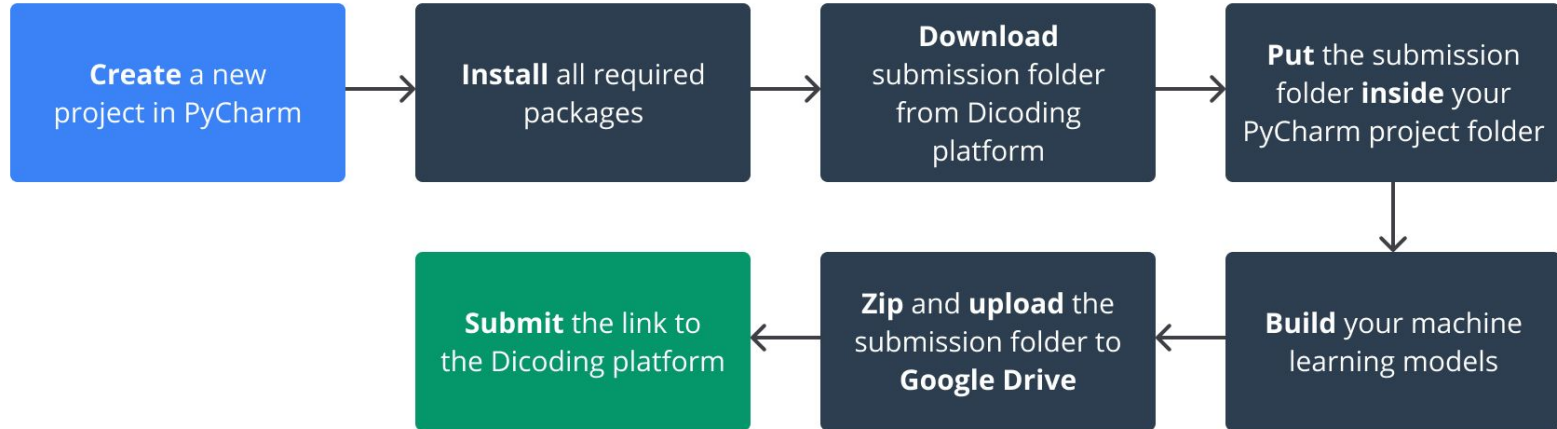
Simulation **Guideline**

- Ensure that your computer meets the system requirement:
 - OS: Windows, Linux, or MacOS
 - Processor: Intel Dual Core (Recommended Core i3 and above)
 - RAM 4GB (Recommended 8GB)
 - Disk Space: 2.5 GB (Recommended minimum to have 5GB of free space)
 - Software: PyCharm IDE
- Python 3.8 installed on your computer
- Undertake this Exam Simulation in PyCharm IDE
- [Setting up your Environment](#)

Simulation **Guideline**

- Exam Time Limit: 5 hours
- Whatever Learning Resources is Allowed
- Complete the Exam on your Own
- **Do not** Share your Exam Code or Talk about the Exam Content with Anyone
- Be Aware that the Size of the Dataset and the Saved Model in the .h5 file extension are Big

Simulation **Workflow**



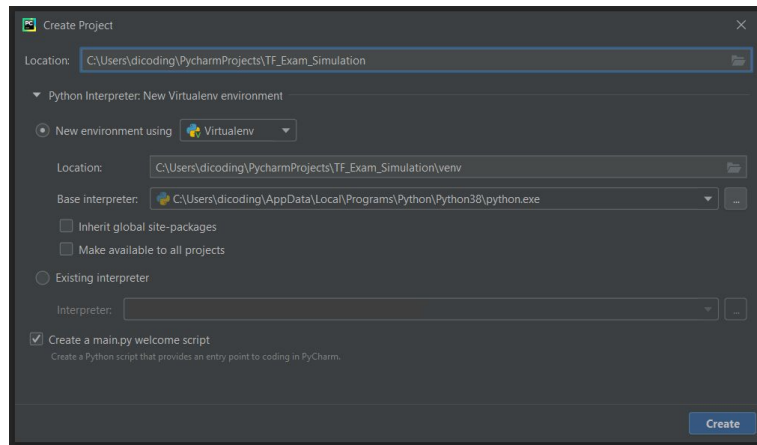
Create a Project in your PyCharm IDE

Open PyCharm and start a new project in PyCharm.

You can name it Exam-Simulation-A or whatever you like.

Make sure that the Base Interpreter is **Python 3.8** as shown in the picture.

We don't recommend you to use Python Anaconda as your base interpreter since the real TF exam does not support this.



Install All Required Packages

You need to install all required libraries (the same libraries used in a real TF Exam Simulation):

- TensorFlow,
- TensorFlow-dataset,
- NumPy,
- pillow,
- urllib3

Refer to this document: [Setting Environment](#) for the detailed installation guide.

Download the Submission Folder

- The simulation exam packages are inside a folder (Submission Folder), i.e Submission A, Submission B, and Submission C.
- This folder contains the starter code for each problem.
- Download the folder to your machine.
- Place the submission folder inside your PyCharm project folder

Build your Machine Learning Model

- Open each starter code in the submission folder
- Build a machine learning model according to the instructions
- If your machine learning model is built successfully, it will save your model as a .h5 file
- This .h5 file model is automatically saved in the same folder with your .py file

Compressed (ZIP) the Submission Folder

- Select all of 10 files (5 of code.py and 5 of model.h5)
- Right-click on the Files, then point to Send to and Select Compressed (zipped) folder.
- Your folder structure must be like this:

Submission_xx.zip

- - Problem_xx.py
- - Model_xx.h5
- ...
- - Model_xx.h5
- Upload the Submission Folder to your Google Drive
- Then Submit the Link to Dicoding Platform

Submission Checklist

- Submission link **could be accessed** by our reviewer teams
- Submission is in **ZIP format**
- Submission folder contains **5 of .py codes and 5 of .h5 models** (1 code and 1 model for each problem)
- Meet the requirement of **accuracy metrics in each problems**
- **Follow the instructions** of each problem

Exam Tips

Example Problem 1

```
# PROBLEM 1
#
# Given two arrays, train a neural network model to
# match the X to the Y.
# Predict the model with new values of X [-2.0, 10.0]
# We provide the model prediction, do not change the
# code.
#
# The test infrastructure expects a trained model that
# accepts
# an input shape of [1].
# Do not use lambda layers in your model.
#
# Please be aware that this is a linear model.
# We will test your model with values in a range as
# defined in the array to make sure your model is
# linear.
#
# Desired loss (MSE) < 1e-4
```

Metrics: Loss (MSE) < 1e-4

Example Problem 2

```
# PROBLEM 2
#
# Build a classifier for the Fashion MNIST dataset.
# The test will expect it to classify 10 classes.
# The input shape should be 28x28 monochrome. Do not
# resize the data.
# Your input layer should accept (28, 28) as the input
# shape.
#
# Don't use lambda layers in your model.
#
# Desired accuracy AND validation_accuracy > 83%
```

accuracy &
validation_accuracy > 83%

Example Problem 3

```
# PROBLEM 3
#
# Build a CNN based classifier for Cats vs Dogs
dataset.
# Your input layer should accept 150x150 with 3 bytes
color as the input shape.
# This is unlabeled data, use ImageDataGenerator to
automatically label it.
# Don't use lambda layers in your model.
#
# The dataset used in this problem is originally
published in
https://www.kaggle.com/c/dogs-vs-cats/data
#
# Desired accuracy and validation_accuracy > 72%
```

accuracy &
validation_accuracy > 72%

Example Problem 4

```
# PROBLEM 4
#
# Build and train a classifier for the sarcasm
dataset.
# The classifier should have a final layer with 1
neuron activated by sigmoid.
#
# Do not use lambda layers in your model.
#
# Dataset used in this problem is built by Rishabh
Misra (https://rishabhmisra.github.io/publications).
#
# Desired accuracy and validation_accuracy > 75%
```

accuracy &
validation_accuracy > 75%

Example Problem 5

```
# PROBLEM 5
#
# Build and train a neural network model using the
Daily Min Temperature.csv dataset.
# Use MAE as the metrics of your neural network model.
# We provided code for normalizing the data. Please do
not change the code.
# Do not use lambda layers in your model.
#
# The dataset used in this problem is downloaded from
https://github.com/jbrownlee/Datasets
#
# Desired MAE < 0.19 on the normalized dataset.
```

MAE < 0.19 on the
normalized dataset

Sharing Session

Discussions

Quiz

Thank You