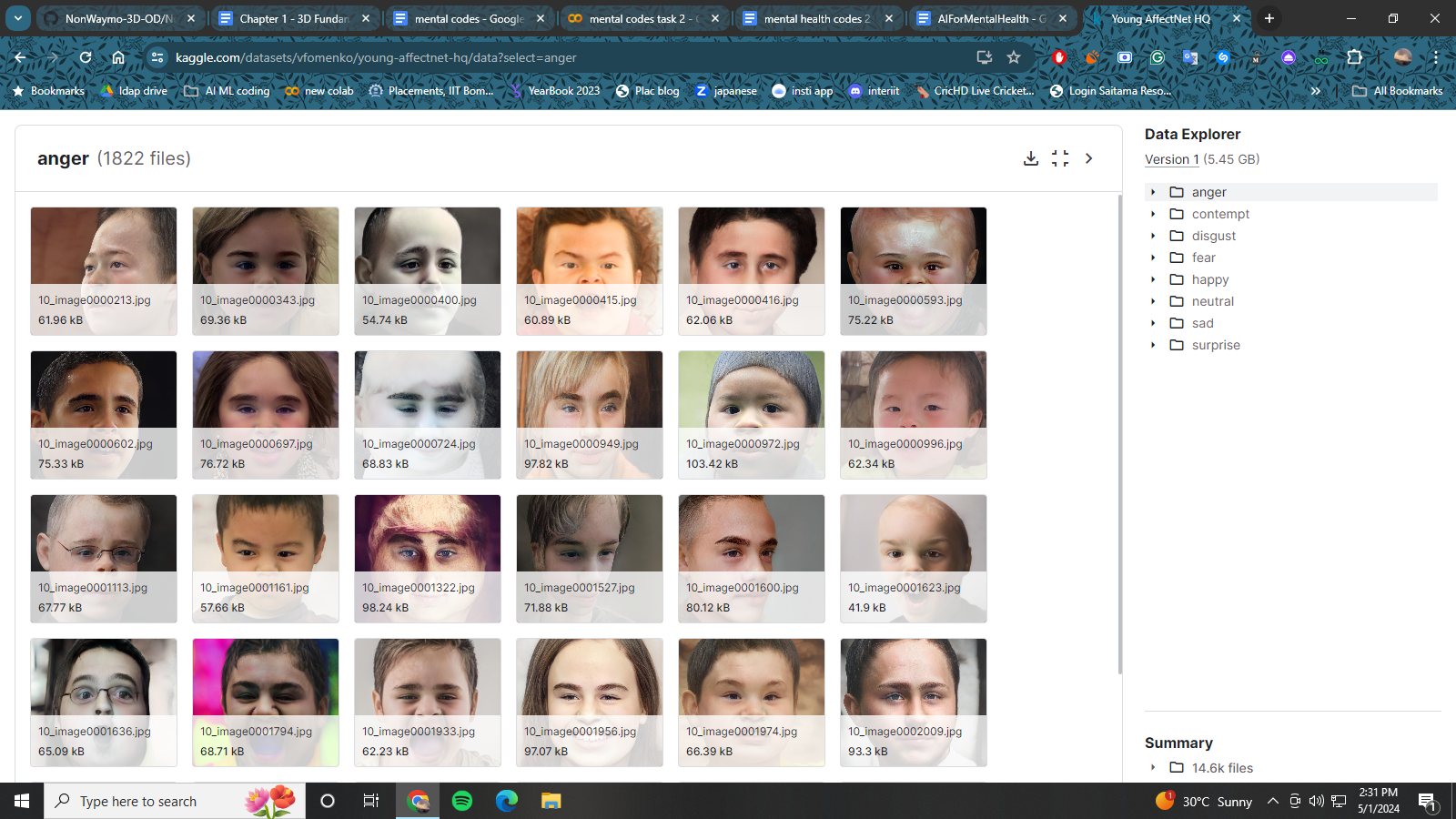
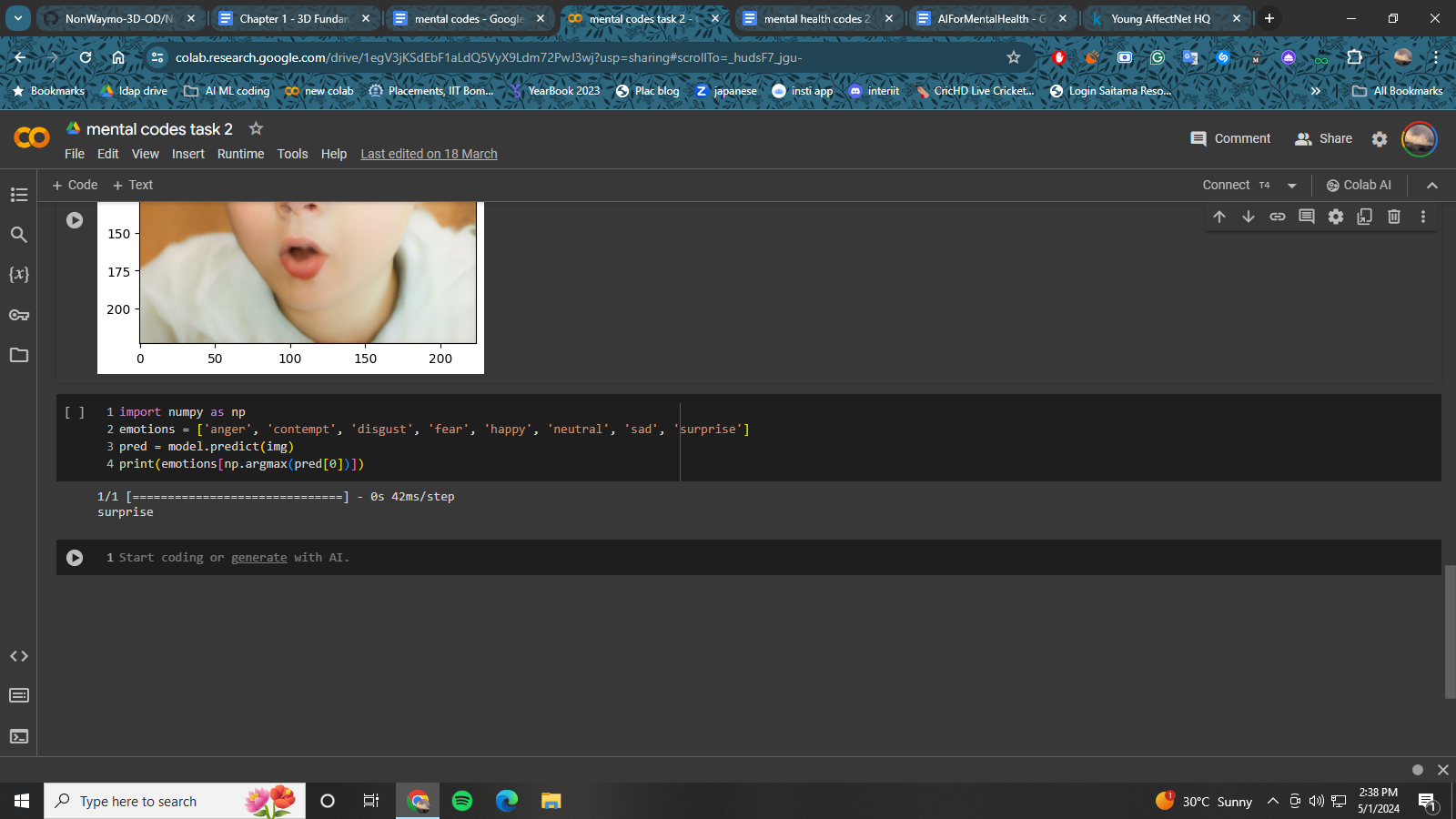
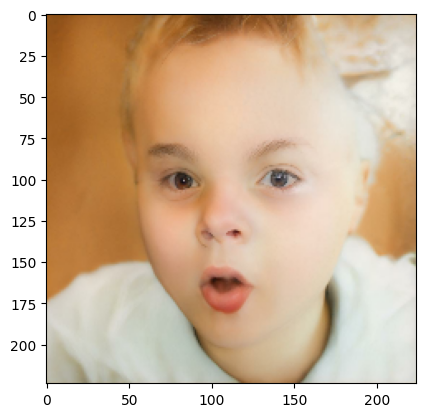
# AI to detect mood changes from images

In this part we take popular facial emotion dataset- [Affectnet](https://www.kaggle.com/datasets/vfomenko/young-affectnet-hq) and perform emotion recognition to detect mood changes and patterns from the facial images. The dataset consists of images for 8 different emotional classes namely- anger, contempt, disgust, fear, happy, neutral, sad, surprise with more than 1500 images for each emotion.



We will treat this as a simple image classification .model where we will use a popular EfficientNetV2 pretrained on imagenet dataset. EfficientNetV2 is a very popular medium sized model which has proved its proficiency on various benchmarks. We will also use the KerasCV library which makes the popular models accessible at one place and provides ease of access. Keras library is easy to use for dataset preparation and finetuning the model.



## CODE:

## Installation

Pip install keras keras-cv

## Imports

import keras

import keras\_cv

Import numpy as np

## Data Loading

train\_ds = keras.utils.image\_dataset\_from\_directory(

directory="data/", labels='inferred',

label\_mode='categorical', batch\_size=32,

image\_size=(224, 224))

## Model Building

backbone = keras\_cv.models.EfficientNetV2Backbone.from\_preset(

"efficientnetv2\_b0\_imagenet")

model = keras\_cv.models.ImageClassifier(backbone=backbone,

num\_classes=8, activation="softmax",)

model.compile(loss='categorical\_crossentropy',

optimizer=keras.optimizers.Adam(learning\_rate=1e-5),

metrics=['accuracy'])

## Model fit

model.fit(train\_ds, epochs=3)

## Model prediction

emotions = ['anger', 'contempt', 'disgust', 'fear', 'happy', 'neutral', 'sad', 'surprise']

pred = model.predict(img)

print(emotions[np.argmax(pred[0])])