```
import
os
        import random
        import string
        from pathlib import Path
        import numpy as np
        from tensorflow.keras.models import load_model
        from PIL import Image, ImageOps
        def random_name_generator(n: int) -> str:
                """
                Generates a random file name.
                Args:
                        n (int): Length the of the file name.
                Returns:
                        str: The file name.
                """
                return ''.join(random.choices(string.ascii_uppercase + string.digits,
        k=n))
        def recognize(image: bytes) -> tuple:
                """
                Predicts the digit in the image.
                Args:
                        image (bytes): The image data.
                Returns:
                        tuple: The best prediction, other predictions and file name
                """

                model=load_model(Path("./model/model.h5"))
                img = Image.open(image).convert("L")

                # Generate a random name to save the image file.
                img_name = random_name_generator(10) + '.jpg'
                if not os.path.exists(f"./static/data/"):
                        os.mkdir(os.path.join('./static/', 'data'))
                img.save(Path(f"./static/data/{img_name}"))
                # Convert the Image to Grayscale, Invert it and Resize to get better
        prediction.
                img = ImageOps.grayscale(img)
                img = ImageOps.invert(img)
                img = img.resize((28, 28))
                # Convert the image to an array and reshape the data to make prediction.
                img2arr = np.array(img)
                img2arr = img2arr / 255.0
                img2arr = img2arr.reshape(1, 28, 28, 1)

                results  = model.predict(img2arr)
```

```
best = np.argmax(results,axis = 1)[0]

# Get all the predictions and it's respective accuracy.
pred = list(map(lambda x: round(x*100, 2), results[0]))

values = [0, 1, 2, 3, 4, 5, 6, 7, 8, 9]
others = list(zip(values, pred))

# Get the value with the highest accuracy
best = others.pop(best)
return best, others, img_name
```