**Server.js**

import path from 'path'

import express from 'express'

import dotenv from 'dotenv'

import colors from 'colors'

import morgan from 'morgan'

import { notFound, errorHandler } from './middleware/errorMiddleware.js'

import connectDB from './config/db.js'

import productRoutes from './routes/productRoutes.js'

import userRoutes from './routes/userRoutes.js'

import orderRoutes from './routes/orderRoutes.js'

import uploadRoutes from './routes/uploadRoutes.js'

dotenv.config()

connectDB()

//The below line of code initiates the express application by assigning the app variable to it.

const app = express()

if (process.env.NODE\_ENV === 'development') {

  app.use(morgan('dev'))

}

colors.enable();

app.use(express.json())

app.use('/api/products', productRoutes)

app.use('/api/users', userRoutes)

app.use('/api/orders', orderRoutes)

app.use('/api/upload', uploadRoutes)

app.get('/api/config/paypal', (req, res) =>

  res.send(process.env.PAYPAL\_CLIENT\_ID)

)

const \_\_dirname = path.resolve()

app.use('/uploads', express.static(path.join(\_\_dirname, '/uploads')))

if (process.env.NODE\_ENV === 'production') {

  app.use(express.static(path.join(\_\_dirname, '/frontend/build')))

  app.get('\*', (req, res) =>

    res.sendFile(path.resolve(\_\_dirname, 'frontend', 'build', 'index.html'))

  )

} else {

  app.get('/', (req, res) => {

    res.send('API is running....')

  })

}

app.use(notFound)

app.use(errorHandler)

const PORT = process.env.PORT || 5000

app.listen(

  PORT,

  console.log(

    `Server running in ${process.env.NODE\_ENV} mode on port ${PORT}`.yellow.bold

  )

)