Next.js - TypeScript Support

Next.js, has execellent support for typescript. Following are few steps to enabling typescript in project.

Create tsconfig.json

Create tsconfig.json in root directory. We're keeping it empty initially. Now start the server.

Next.JS will detect tsconfig.json and show follwing message on console.

```
npm run dev

> nextjs@1.0.0 dev D:\Node\nextjs
> next

ready - started server on http://localhost:3000
It looks like you're trying to use TypeScript but do not have the required

Please install typescript, @types/react, and @types/node by running:

        npm install --save-dev typescript @types/react @types/node

If you are not trying to use TypeScript, please remove the tsconfig.json f
...
```

Install typescript

Run the npm install command to install typescript and related libraries.

```
npm install --save-dev typescript @types/react @types/node
...

+ @types/node@14.0.23
+ @types/react@16.9.43
+ typescript@3.9.6
added 5 packages from 72 contributors and audited 839 packages in 27.538s
...
```

Start Next.js Server

Run the following command to start the server −.

```
npm run dev
```

```
> nextjs@1.0.0 dev D:\Node\nextjs
> next

ready - started server on http://localhost:3000
We detected TypeScript in your project and created a tsconfig.json file fo


Your tsconfig.json has been populated with default values.

event - compiled successfully
wait  - compiling...
event - compiled successfully
```

Open tsconfig.json

NextJS server has modified the tsconfig.json.

```
{
   "compilerOptions": {
      "target": "es5",
      "lib": [
         "dom",
         "dom.iterable",
         "esnext"
      ],
      "allowJs": true,
      "skipLibCheck": true,
      "strict": false,
      "forceConsistentCasingInFileNames": true,
      "noEmit": true,
      "esModuleInterop": true,
      "module": "esnext",
      "moduleResolution": "node",
      "resolveJsonModule": true,
      "isolatedModules": true,
      "jsx": "preserve"
   },
   "exclude": [
      "node_modules"
   ],
   "include": [
      "next-env.d.ts",
      "**/*.ts",
      "**/*.tsx"
   ]
}
```

Create hello.ts

Create hello.ts in pages/api directory which will acts as a rest service for us.

```
import { NextApiRequest, NextApiResponse } from 'next'

export default (_: NextApiRequest, res: NextApiResponse) => {
   res.status(200).json({ text: 'Welcome to TutorialsPoint' })
}
```

Start Next.js Server

Run the following command to start the server −.

```
npm run dev
> nextjs@1.0.0 dev \Node\nextjs
> next

ready - started server on http://localhost:3000
event - compiled successfully
event - build page: /
wait  - compiling...
event - compiled successfully
event - build page: /next/dist/pages/_error
wait  - compiling...
event - compiled successfully
```

Verify Output

Open localhost:3000/api/hello in a browser and you will see the following output.

```
{"text":"Welcome to TutorialsPoint"}
```