

## TYPE SCRIPT LANGUAGE

### Introduction of Type Script:

Type Script is a programming language that is a superset of JavaScript. This means that any valid JavaScript code is also valid Type Script code, but Type Script extends JavaScript by adding static typing, interfaces, and other features. It is designed to make it easier to develop large-scale applications by providing developers with tools for catching common errors during development.

ٹائپ اسکرپٹ ایک پروگرامنگ زبان ہے جو جاوا اسکرپٹ کا ایک سپر سیٹ ہے۔ اس کا مطلب یہ ہے کہ کوئی بھی درست جاوا اسکرپٹ کوڈ بھی درست ٹائپ اسکرپٹ کوڈ ہے، لیکن ٹائپ اسکرپٹ جامد ٹائپنگ، انٹرفیس اور دیگر خصوصیات شامل کر کے جاوا اسکرپٹ کو بڑھاتا ہے۔ یہ ترقی کے دوران عام غلطیوں کو پکڑنے کے لئے ڈویلپرز کو تولز فراہم کر کے بڑے پیمانے پر ایپلی کیشنز تیار کرنا آسان بنانے کے لئے ڈیزائن کیا گیا ہے۔

ہذا یعنی اُن اُی کوڈ جافا سکرپٹ صالح ہو اُیضا رمز نوع نصی صالح، لیکن JavaScript ہی لغة برمجة تمثل مجموعة شاملة من Type Script عن طريق إضافة كتابة ثابتة وواجهات وميزات أخرى. إنه مصمم لتسهيل تطوير التطبيقات واسعة النطاق من JavaScript بوسع Type Script خلال تزويد المطورين بأدوات لالتقاط الأخطاء الشائعة أثناء التطوير.

### Purpose:

The main purpose of Type Script is to enhance the development experience by introducing static typing. This allows developers to catch type-related errors early in the development process, reducing bugs and improving code quality. Additionally, Type Script provides features like interfaces and advanced tooling that make it easier to work with larger codebases.

ٹائپ اسکرپٹ کا بنیادی مقصد جامد ٹائپنگ متعارف کروا کر ترقیاتی تجربے کو بہتر بنانا ہے۔ اس سے ڈویلپرز کو ترقی کے عمل میں ابتدائی طور پر قسم سے متعلق غلطیوں کو پکڑنے، غلطیوں کو کم کرنے اور کوڈ کے معیار کو بہتر بنانے کی اجازت ملتی ہے۔ مزید برآں، ٹائپ اسکرپٹ انٹرفیس اور جدید ٹولنگ جیسی خصوصیات فراہم کرتا ہے جو بڑے کوڈ بیس کے ساتھ کام کرنا آسان بناتے ہیں۔

ہو تعزیر تجربه التطوير من خلال إدخال الكتابة الثابتة. يتيح ذلك للمطورين اكتشاف Type Script الغرض الرئيسي من الأخطاء المتعلقة بالنوع في وقت مبكر من عملية التطوير، وتقليل الأخطاء وتحسين جودة التعليمات البرمجية. بالإضافة إلى ميزات مثل الواجهات والأدوات المتقدمة التي تسهل العمل مع قواعد التعليمات البرمجية الأكبر Type Script ذلك، يوفر

### Why Type Script:

1. **Static Typing:** Type Script introduces static typing, which helps catch errors before runtime.
2. **Improved Tooling:** Type Script offers better development tools, including code navigation and auto completion.
3. **Readability and Maintainability:** With the addition of types and interfaces, code becomes more self-documenting and easier to understand.
4. **Compatibility with JavaScript:** Type Script is a superset of JavaScript, allowing developers to gradually adopt it in existing projects.

جامد ٹائپنگ: ٹائپ اسکرپٹ جامد ٹائپنگ متعارف کرواتا ہے، جو رن ٹائم سے پہلے غلطیوں کو پکڑنے میں مدد کرتا ہے۔ بہتر ٹولنگ: ٹائپ اسکرپٹ بہتر ترقیاتی تولز پیش کرتا ہے، بشمول کوڈ نیویگیشن اور آٹو تکمیل۔ پڑھنے کی قابلیت اور برقرار رکھنے کی صلاحیت: اقسام اور انٹرفیس کے اضافے کے ساتھ، کوڈ زیادہ خود دستاویزی اور سمجھنے میں آسان ہوجاتا ہے۔ جاوا اسکرپٹ کے ساتھ مطابقت: ٹائپ اسکرپٹ جاوا اسکرپٹ کا ایک سپر سیٹ ہے، جس سے ڈویلپرز کو موجودہ منصوبوں میں آہستہ آہستہ اسے اپنانے کی اجازت ملتی ہے۔

أدوات تطوير أفضل، Type Script الكتابة الثابتة، مما يساعد على اكتشاف الأخطاء قبل وقت التشغيل. أدوات محسنة: يوفر Type Script الكتابة الثابتة: يقدم بما في ذلك التنقل في التعليمات البرمجية والإكمال التلقائي. سهولة القراءة والصيانة: مع إضافة الأنواع والواجهات، تصبح التعليمات البرمجية أكثر توثيقا ذاتيا، مما يسمح للمطورين باعتمادها تدريجيا في المشاريع الحالية JavaScript عبارة عن مجموعة شاملة من JavaScript: Type Script وأسهل في الفهم. التوافق مع

**Usage and Application:** Type Script is commonly used for:

1. **Web Development:** Especially for larger and more complex web applications.
2. **Node.js Development:** Type Script can be used for server-side development with Node.js.
3. **Cross-Platform Development:** Type Script is used in frameworks like Angular for building cross-platform applications.

ترقی: ٹائپ اسکریپٹ js. ویب ڈویلپمنٹ: خاص طور پر بڑے اور زیادہ پیچیدہ ویب ایپلی کیشنز کے لئے. نوڈ کے ساتھ سرور سائیڈ ڈویلپمنٹ کے لئے استعمال کیا جاسکتا ہے۔ کراس پلیٹ فارم ڈویلپمنٹ: js. کو نوڈ ٹائپ اسکریپٹ کراس پلیٹ فارم ایپلی کیشنز کی تعمیر کے لئے اینگولر جیسے فریم ورک میں استعمال ہوتا ہے۔

يمكن استخدام js. لتطوير الويب: خاصة لتطبيقات الويب الأكبر والأكثر تعقيدا. تطوير العقدة التطوير عبر الأنظمة الأساسية: js. Node.js. للتطوير من جانب الخادم باستخدام Type Script. لبناء تطبيقات عبر الأنظمة الأساسية Angular في أطر عمل مثل Type Script يتم استخدام.

#### **Main Differences with JavaScript:**

1. **Static Typing:** Type Script is statically typed, meaning that variable types are declared at compile-time, whereas JavaScript is dynamically typed.
2. **Interfaces:** Type Script supports interfaces, allowing developers to define contracts for object shapes. JavaScript lacks native support for interfaces.
3. **Compilation:** Type Script code needs to be transpiled (converted) to JavaScript before running in the browser or on a server, while JavaScript is interpreted directly by the browser or the Node.js runtime.
4. **Tooling Support:** Type Script provides a richer set of development tools due to its static nature, offering better code navigation, auto completion, and refactoring support

جامد ٹائپنگ: ٹائپ اسکریپٹ جامد طور پر ٹائپ کیا جاتا ہے ، جس کا مطلب ہے کہ متغیر اقسام کو کمپائل ٹائم پر اعلان کیا جاتا ہے ، جبکہ جاوا اسکریپٹ متحرک طور پر ٹائپ کیا جاتا ہے۔ انٹرفیس: ٹائپ اسکریپٹ انٹرفیس کی حمایت کرتا ہے ، جس سے ڈویلپر کو آبجیکٹ کی شکلوں کے لئے معاہدوں کی وضاحت کرنے کی اجازت ملتی ہے۔ جاوا اسکریپٹ میں انٹرفیس کے لئے مقامی حمایت کا فقدان ہے۔ تالیف: ٹائپ اسکریپٹ کوڈ کو براؤزر میں یا سرور پر چلانے سے پہلے جاوا اسکریپٹ میں منتقل (تبدیل) کرنے کی ضرورت ہوتی ہے ، جبکہ رن ٹائم کے ذریعہ تشریح کی جاتی ہے۔ ٹولنگ سپورٹ: ٹائپ اسکریپٹ اپنی جامد نوعیت js. جاوا اسکریپٹ کی براہ راست براؤزر یا نوڈ کی وجہ سے ترقیاتی ٹولز کا ایک امیر سیٹ فراہم کرتا ہے ، جو بہتر کوڈ نیویگیشن ، آٹو تکمیل ، اور ریفیکٹرنگ سپورٹ پیش کرتا ہے۔

بشکل ثابت ، مما يعني أنه يتم الإعلان عن أنواع المتغيرات في وقت التحويل البرمجي ، Type Script الكتابة الثابتة: تتم كتابة الواجهات، مما يسمح للمطورين بتحديد عقود أشكال Type Script ديناميكيا. واجهات: يدعم JavaScript بينما يتم كتابة الكائنات. تفتقر جافا سكريبت إلى الدعم الأصلي للواجهات. جميع: نوع يجب تحويل شفرة البرنامج النصي (تحويلها) إلى مباشرة بواسطة المتصفح أو وقت تشغيل JavaScript قبل تشغيلها في المتصفح أو على الخادم ، بينما يتم تفسير JavaScript مجموعة أكثر ثراء من أدوات التطوير نظرا لطبيعته الثابتة ، مما يوفر تنقلا أفضل Type Script دعم الأدوات: يوفر Node.js في التعليمات البرمجية ، والإكمال التلقائي ، ودعم إعادة البناء