1. What are the OOPs concepts that are supported in typescript ?

Ans: Typescript supports almost all the OOPs concepts, class, object, Inheritance, encapsulation, overriding, abstraction.

But typescript doesnot allows overloading.

_____1

2. Why typescript does not allows overloading?

Ans: JavaScript does not allow overloading, so TypeScript cannot generate multiple definitions of the same function

differing only by their signature. TypeScript does not accept the two different constructors too(constructor overloading).

_____2

3. How to define constructors in typescript ?

Ans: To define constructor in typescript "constructor" is the keyword that should be used. Unlike languages like c#/ java

where constructor name should be same as class name, constructor should not be same as class name in typescript.

Ex:

```
class Sample(){

let firstName: string
```

```
let lastName: string
constructor(fn: string,ln: string){
        this.firstName = fn;
        this.lastName = ln;
  }
}
```

4. What are the types of typecasting that are supported in typescript ?

Ans: typescript allows only explicit typecasting by using < > or as keyword.

_____4

5. What is type assertion in typescript ?

Ans: Overiding/ customizing the view of types into anyway you want to do is know as type assertion mechanism.

This mechanism tells the compiler that you can handle the types better that it does.

_____5

6. Why Type assertion is considered as harmfull ?

Ans: When an object have zero properties by using Type Assertion we can handle that issues, but the most hazardious thing is

compiler donot throws any error.But the compiler provides autocomplete for properties of the object.

_____6

7. What is Double Assertion ?

Ans: As type assertion is a bit unsafe you can use double assertion, by using 'any' which is compatible with all the types

compiler donot throws errors.

Ex: let str = num as any as string;

_____7

8. Differences between type casting and type assertion ?

Ans:

Type casting: It generally implies some sort of runtime support.

Tyep Assertion: It is purely a compile time construct.

_____8

9. What are abstract classes in typescript ?

Ans: Abstract classes are base classes from which other classes may be derived and are incomplete/partially implemented classes.

If you need a default implementation for a method then you can choose abstract otherwise we can go for interface which is

purely abstract.

A member function declared abstract does not have an implementation.

_____9

10. What are the advantages of using Abstract classes in typescript ?

Ans:

1. Programmers can write partial implementaion of classes.
2. abstract annotations clarify the role of base classes in various patterns.
3. The programmers will be warned about the classes that shouldn't be initiated.