What is gRPC:

1. gRPC is open source RPC (Remote Procedure Calls) developed by Google
2. At a high level, gRPC allows us to define REQUEST & RESPONSE for RPC (Remote Procedure Calls) and handle all the rest for us.
3. It is modern, fast (approx 25 times than REST), efficient (Uses protobufs)
4. It is build on a top of HTTP/2 (HTTP/2 have significant advantage over current HTTP/1.1 Implementation in REST)

Advantages of gRPC:

1. It uses Protocol Buffers. A single proto file works over multiple programming language (including client & server) and allow to use a framework that scales to millions of RPC per second)
2. Very convenient for transporting lot of data
3. Protocol buffers defines rules to make API evolve without breaking existing client which is useful in microservices
4. Size of protocol buffer is lesser than JSON hence we save lot of network bandwidth

Advantages of HTTP/2:

1. HTTP/2 supports multiplexing (this reduces latency).
2. It also allows server push (Server can push stream for one request to client).
3. It allows header compressions. HTTP/2 is binary and by default SSL enabled (Secured)

gRPC vs HTTP API with JSON:

Feature	gRPC	HTTP API with JSON
Contract	Required (*.proto*)	Optional (OpenAPI, Swagger)
Protocol	HTTP/2	HTTP
Payload	Protobuf (small, binary)	JSON (large, human readable)
Prescriptiveness	Strict specification	Loose. Any HTTP is valid.
Streaming	Client, server, bi-directional	Client, server
Browser Support	No (Requires grpc-web)	Yes
Client code-generation	Yes	OpenAPI or third party tooling