*فانکشن اول یک promise ایجاد می کند که کار خودش را بعد از مدت زمانی که به عنوان ارگومان ورودی (interval ) داده شده است است مثلا بعد از 5000 یا 3000 میلی ثانیه کار خودش را با موفقیت به پایان میرساند.*

return new Promise((resolve, reject) => {

setTimeout(function(){

resolve("successful");

}, interval);

});

};

*فانکشن دوم یک promise ایجاد می کند که کار خودش را بعد از مدت زمانی که به عنوان ارگومان ورودی (interval ) داده شده است است مثلا بعد از 2000 میلی ثانیه با شکست اجرا می شود و کد بخش catch اجرا می شود.*

function timeoutPromiseReject(interval) {

return new Promise((resolve, reject) => {

setTimeout(function(){

reject("error");

}, interval);

});

};

*فانکشن سوم دو فانکشن اول صدا زده شده است.*

*startTime تعداد میلی ثانیه ها را از اول ژانویه 1970 برمی گرداند..*

*قسمت catch بعد از 7000 میلی ثانیه اجرا میشه به دلیل اینکه* timeoutPromiseResolve(5000);

*با موفقیت تموم میشه و متود then اجرا میشه و وارد catch نمیشه به محض انکه کار timeoutPromiseReject(2000) تمام شد و با شکست واجه شد متود catch اجرا میشه* و در واقع به finishTime ، 7000 اضافه شده است.

یعنی timeTaken برابر با 7000 میلی ثانیه است.