

- 1) playVideo(player)
- 2) pauseVideo(player)
- 3) loadVideo(player,videoId)
- 4) cueVideo(player,videoId)
- 5) loadPlaylist(player,playlistId,index) //playlistId can be either array of video Ids or string of a playlist Id
- 6) showCurrentTime(player,span)
- 7) twoDigit(num)
- 8) var timers=new Array();//one array for one player  
eventsGivenTime(player,timers,videoId, [  
    { time: 5, funct: function () { } },  
    { time: 8, funct: function () { } },  
    { time: 15, funct:function () { } }  
]);
- 9) timeBarSetUp(slider, video)
- 10) changeTime(player,value,max,spanDisplayTime)
- 11) eventForIrregularImg(canva, imgSrc, mouseEvent, funct)
- 12) sliderBarMode(event,div,actionDrag,actionEnd,axis)
  - a) event target can be anything, but it's always element with class :“handle” dragged in div.
  - b) need to assign dynamic position to div(position:relative, position:absolute...etc)
  - c) If axis==“y”, it behaves like vertical bar, otherwise horizontal bar.
  - d) function actionDrag(value, max, cleardrag){}
    - i) use “cleardrag();” when you want to cleardrag when without mouseup. It is used in progressBarAndTimeSpanSetUp(player,div,timeSpan).
  - e) function actionEnd(value, max){}
- 13) clickOnSliderBar(event,div,actionClick,axis)
  - a) event target is the bar
  - b) need to assign dynamic position to div(event target)(position:relative, position:absolute...etc)
  - c) function actionClick(value,max){}
- 14) mouseEnterSliderBar(event,actionHover,acionLeave,actionEnter)
  - a) event target is the bar
  - b) need to assign dynamic position to div(event target) (position:relative, position:absolute...etc)
- 15) updateSliderBarPosition(value, max, div,axis)
  - a) update slider bar according to value and max
  - b) need to assign dynamic position to div(position:relative, position:absolute...etc)
- 16)
- 17) findDocumentCoords(mouseEvent)
- 18) mouseCoordsToElem(div,mouseEvent)
- 19) showThumbNail(videoID,div)
- 20) videoInfo(videoId,funct)//need google API key
- 21) playlistInfo(playlistId,funct)//need google API key
- 22) fullScreen(playerDivId)
- 23) setList()
  - a) setList({

```

valueArray: trackList.code,
nameArray: trackList.name,
listDiv: divId,
actionNull:function(){},
actionExist:function(){},
actionChange: function(selectedListItem,value){}
});

```

b) two different kind of track list

i) `<select id=" divId ">`

```

    <option class="sampleNode nameSpan" style="display:block;">select
    language</option>

```

`</select>`

ii) `<div id=' divId ' style="display:inline-block; max-height:8em; overflow-y:scroll;">`

```

    <p class="sampleNode"><input type="radio" style="display:none;"><label style="font-
    size:0.8em; line-height:100%; display:none;" class=" nameSpan "></label></p>

```

c) need to set up a not displayed section in htm, which className is “sampleNode”, with dissidents of class:nameSpan

d) used In setTracklist and setPlaybackRateList

24) getTrack(player,callback)

a) `callback( {value: languageCode[], name: languageName[]})`

25) setListForVideoAsync(player,div,defaultValue,essentialFuncnt,styling)

a) Full format(use return in actionSetUp):

```

setListForVideo (player1, tracks2, 'en',
{
    actionSetUp: function (player,wrapper) { asyncFuncnt(player,wrapper/*callback*/); },
    actionChange: function (player, value) { },
    actionNull:function(){ },//styling when the list exists or not
    actionExist:function(){ }
},
{
    stylingFuncnt:[
        { event: "mouseenter", funct: function (event) { } },
        { event: "mouseleave", funct: function (event) { } }
    ],
    styleSelected: function(item_to_be_styled) { },
    styleUnselected: function(item_to_be_styled) { }
}
);

```

b) Use async function(ex:getTrack) in actionSetUp:

In getTrack, callback must have input:{ value:array, name:array }

`setListForVideo (player1, tracks2, 'en',`

```

{
    actionSetUp: function (player,wraper) { getTrack(player,wraper);},
    actionChange: function (player, value) { }
}

```

- ```
});
```
- c) Reload “on state change “and “video changed”
  - d) need to set up a not displayed section in htm, which className is “sampleNode”, with dissidents of class:nameSpan
  - e) defaultValue doesn’t do anything. It has to be the same as youtube default to make sense.
  - f) two different kind of track list(see setList())
  - g) For input object styling, better using css styling
  - h) In stylingFunct, event.target is label or list item
  - i) In stylingFunct, can specify arbitrary events(mousemove, click, mouseenter,...etc) for arbitrary times.
  - j) In styleSelected, item\_to\_be\_styled is either checked label or the selected list item.
  - k) In styleUnselected, item\_to\_be\_styled is either unchecked label or the unselected list items.
  - l) Example: caption language

```

setListForVideoAsync(player1, v.tracklist, 'en',
{
  actionSetUp: function (player, wrap) { getTrack(player, wrap); },
  actionChange: function (player, value) {
    player.setOption("captions", "track", { "languageCode": value });
    player.setOption("cc", "track", { "languageCode": value });
    v.showLan.innerHTML = "▼" + document.getElementById("tracklist_" +
value).nextElementSibling.innerHTML;
    v.tracklist.style.display = "none";
  },
  actionNull: function () { v.tracklistNow.parentNode.style.display = "none"; },
  actionExist: function () { v.tracklistNow.parentNode.style.display = "block"; }
});

```

## 26) setListForVideoSync(player,div,defaultValue,essentialFunct,styling)

- a) Full format(use return in actionSetUp):

```

setListForVideo (player1, tracks2, 'en',
{
  actionSetUp: function (player) { return { value: valueArray, name: nameArray }; },
  actionChange: function (player, value) { },
  actionNull:function(){ },//styling when the list exists or not
  actionExist:function(){ }
},
{
  stylingFunct:[
    { event: "mouseenter", funct: function (event) { } },
    { event: "mouseleave", funct: function (event) { } }
  ],
  styleSelected: function(item_to_be_styled) { },
  styleUnselected: function(item_to_be_styled) { }
}

```

);

b) Auto check if video Id or list length is changed on player state change to decide if reload the list.

c) Example: PlayBackRate:

```
setListForVideoSync(player1, v.ratelist, 1,
{
    actionSetUp: getSpeed,
    actionChange: function (player, value) { player.setPlaybackRate(value); v},
    actionNull: function () { v.ratelistNow.parentNode.style.display = "none"; },
    actionExist: function () { v.ratelistNow.parentNode.style.display = "block"; }
});
function getSpeed(player) {
    var speed = player.getAvailablePlaybackRates();
    var name = new Array();
    for (var i = 0; i < speed.length; i++) {
        name.push(speed[i].toString());
        if (speed[i] == 1) name[i] = "normal";
    }
    return { value: speed, name: name };
}
```

d) Example: Quality(not recommended, not entirely correct):

```
setListForVideo(player1, v.qualityList1, 'auto', //!!!!!!!!!!!!not working well because of API's
setPlaybackQuality
{
    actionSetUp: function (player) { var qualities = player.getAvailableQualityLevels(); /*only
has value when event=1 or 2*/return { value: qualities, name: qualities }; },
    actionChange: changeQuality
});
function changeQuality(player, value) {
    player.setPlaybackQuality(value); /*it's only suggesting quality. youtube wil determine the
actual quality by network quality*/
}
```

27) ccon(player,trackList)

28) ccoff(player)

29) getQueryVariable(url,variable)

a) Ex: get video id: getQueryVariable(player.getVideoUrl(),'v')

30) setUpPlaylist({})

```
a) setUpPlaylist({
    player: player1,
    playlistId: 'PLiatqssGebieEqJ_Nx5FQ1CUDe-fybMOw', //Use playlistId, Max 50 items
    div: playlist1,
    actionClick: function (event, index, itemDiv) { },
    actionEnter: function (event, index, itemDiv) { },
    actionLeave: function (event, index, itemDiv) { },
```

```
styleNowPlaying: function(event, nowPlayingDiv){ },  
styleNotPlaying: function(event, notPlayingDiv){ }  
});
```

- b) Need to set up a not displayed section in htm, which className is "sampleNode".
  - c) playlistId can be either array of videoids or a youtube playlistId.  
If using youtube playlist id, max 50 items
  - d) need to set up a not displayed section in htm, with dissidents of  
class:title,channelTitle,thumbnail
- 31) progressBarAndTimeSpanSetUp(player,div,timeSpan)
- a) class: handle, hoverTimeBlock, hoverTimeText
  - b) need to assign dynamic position to div(position:relative, position:absolute...etc)