fixing getStats

object and maplike

Issue 1: Returning a base dictionary returns just that

webidl:

```
interface RTCStatsReport = { getter RTCStats (DOMString id); };
```

js:

```
pc.getStats().then(report => {
  Object.keys(report).forEach(key => {
    var stat = report[key];
    if (stat.type != "candidatepair") continue;

    console.log(stat.timestamp); // 1441898883269.8672
    console.log(stat.priority);  // undefined
    console.log(report[stat.localIceCandidateId].ipAddress); // error
  }
}
```

https://github.com/w3c/webrtc-pc/issues/275

Issue 1: Solutions

webidl:

```
-  interface RTCStatsReport = { getter RTCStats (DOMString id); };
+  interface RTCStatsReport = { getter object (DOMString id); };
```

or

```
-  interface RTCStatsReport = { getter RTCStats (DOMString id); };
+  interface RTCStatsReport = { getter AllRTCStats (DOMString id); };
```

where AllRTCStats would use a new WebIDL feature requested for this *

```
   typedef AllRTCStats = (RTCStats || RTCIceCandidateStats); // etc.
```

*) https://github.com/heycam/webidl/issues/57

Issue 2: getter is legacy tech. Use ES6 maplike:

webidl:

```
  interface RTCStatsReport = { readonly maplike<DOMString, AllRTCStats>; };
```

js:

```
  pc.getStats().then(report => {
   report.forEach(stat => {
      if (stat.type != "candidatepair") continue;

      console.log(stat.timestamp); // 1441898883269.8672
      console.log(stat.priority);  // 7996986662803866000
      console.log(report.get(stat.localIceCandidateId).ipAddress); //192.168.1.1
    }
  }
```

https://github.com/w3c/webrtc-pc/issues/285