1. Create a function expression named createBicyclePrototye which returns an object literal with one property speed set to value 0 and two methods
   1. applyBrake that takes a parameter and decrement the current speed of the object by that given value.
   2. speedup that takes a parameter increment current speed by given value.

|  |
| --- |
| var createBicyclePrototye=(function(){  let speed=0;  function applyBrake(dec){  speed-=dec;  }  function speedup(inc){  speed+=inc;  }  function chvalue(){  return speed;  }  return {  decsped:applyBrake,  incsped:speedup,  tochevale:chvalue  }    })(); |

1. Create another function expression named createMountainBikeProtoype   that take a prototype parameter and returns an object created using Object.create (prototype) with one additional property gear set to value 1 and one additional method setGear which takes a parameter and sets gear value for the object.

|  |
| --- |
| var createMountainBikeProtoype=function(type){  let newobj= Object.create(type);  newobj.gear=7;  newobj.setgear=function(ev){  this.gear=ev;  }  return newobj;  }; |

1. Write third function expression named start that first make bicyclePrototype object using first function and then mountainBikePrototype object using second function, passing bicyclePrototype as argument and play with it (do some console.logs) to see if its working as intended.

|  |
| --- |
| var start=createMountainBikeProtoype(createBicyclePrototye);  start.setgear(5);  console.log(start);  console.log(window.gear); |

1. Now create some objects from both bicyclePrototype and mountainBikePrototype using Object.create() and play with it. (do some console.log)

|  |
| --- |
| var start=createMountainBikeProtoype(createBicyclePrototye);  start.setgear(5);  console.log(start);  console.log(window.gear);  var newo=Object.create(createBicyclePrototye);  newo.decsped(1);  console.log(newo.tochevale());  console.log(newo);  var newo1=Object.create(start);  newo1.setgear(1);  console.log(newo1.tochevale());  console.log(newo1); |

1. Wrap all three functions in a module so that we have only one module variable in the global name space.

|  |
| --- |
| (  function(){    var createBicyclePrototye=(function(){  let speed=0;  function applyBrake(dec){  speed-=dec;  }  function speedup(inc){  speed+=inc;  }  function chvalue(){  return speed;  }  return {  decsped:applyBrake,  incsped:speedup,  tochevale:chvalue  }    })();  createBicyclePrototye.decsped(3);  console.log(createBicyclePrototye.tochevale());  createBicyclePrototye.incsped(4);  console.log(createBicyclePrototye.tochevale());  var createMountainBikeProtoype=function(type){  let newobj= Object.create(type);  newobj.gear=7;  newobj.setgear=function(ev){  this.gear=ev;  }  return newobj;  };  var start=createMountainBikeProtoype(createBicyclePrototye);  start.setgear(5);  console.log(start);  console.log(window.gear);  var newo=Object.create(createBicyclePrototye);  newo.decsped(1);  console.log(newo.tochevale());  console.log(newo);  var newo1=Object.create(start);  newo1.setgear(1);  console.log(newo1.tochevale());  console.log(newo1);  }  )(); |

1. Make use of 'use strict' at the top of your module.

|  |
| --- |
| (    function(){  "use strict";  var createBicyclePrototye=(function(){  let speed=0;  function applyBrake(dec){  speed-=dec;  }  function speedup(inc){  speed+=inc;  }  function chvalue(){  return speed;  }  return {  decsped:applyBrake,  incsped:speedup,  tochevale:chvalue  }    })();  createBicyclePrototye.decsped(3);  console.log(createBicyclePrototye.tochevale());  createBicyclePrototye.incsped(4);  console.log(createBicyclePrototye.tochevale());  var createMountainBikeProtoype=function(type){  let newobj= Object.create(type);  newobj.gear=7;  newobj.setgear=function(ev){  this.gear=ev;  }  return newobj;  };  var start=createMountainBikeProtoype(createBicyclePrototye);  start.setgear(5);  console.log(start);  console.log(window.gear);  var newo=Object.create(createBicyclePrototye);  newo.decsped(1);  console.log(newo.tochevale());  console.log(newo);  var newo1=Object.create(start);  newo1.setgear(1);  console.log(newo1.tochevale());  console.log(newo1);  }  )(); |

1. Now, try to achieve above functionality, this time using ES6 class construct.

|  |
| --- |
|  |