

Rozbor testu

p.2 → [28.02.2019](#)

*Vyjmenujte primitiva (datové typy)
v JavaScript.*

*Vyjmenujte primitiva (**datové typy**)
v JavaScript.*

NE konstruktory



*Vyjmenujte primitiva (**datové typy**)
v JavaScript.*

Object.prototype.toString.call

.constructor

instanceof

NE konstruktory



*Vyjmenujte primitiva (**datové typy**)
v JavaScript.*

~~Object.prototype.toString.call~~

~~.constructor~~

~~instanceof~~

NE konstruktory



*Vyjmenujte primitiva (**datové typy**)
v JavaScript.*



typeof

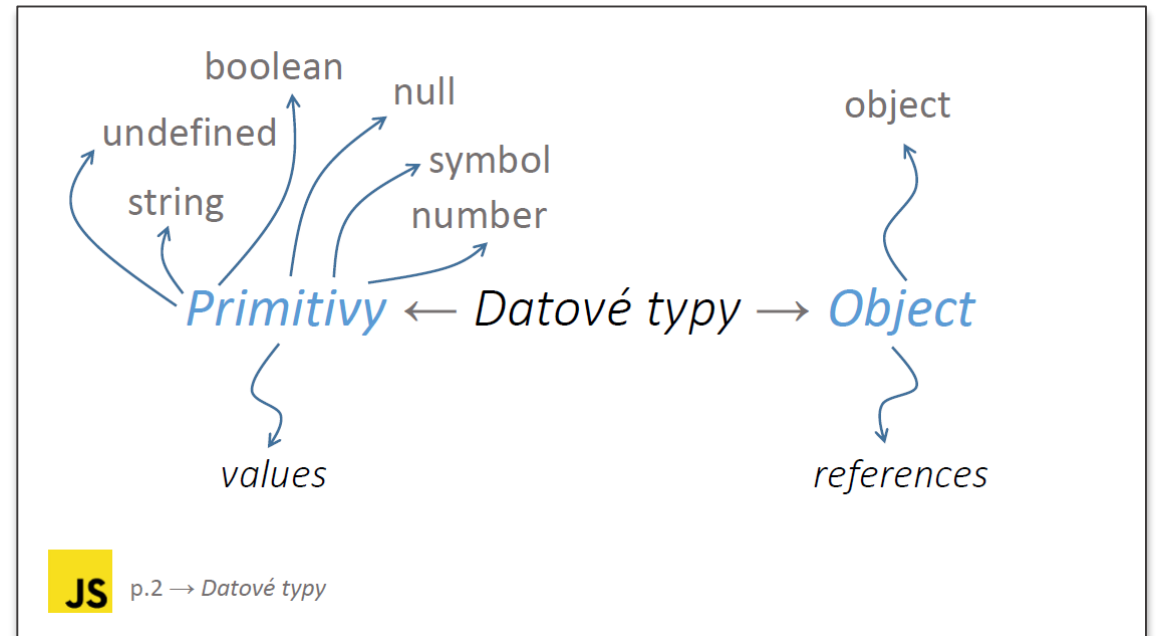
*Vyjmenujte **primitiva** (datové typy)
v JavaScript.*

NE object



*Vyjmenujte **primitiva** (datové typy)
v JavaScript.*

*string, number, boolean,
null, undefined, symbol,
~~object~~*



*Napište funkci **isObject()** která vrací **true** pokud argument je objekt, **false** pokud je primitiv.*

typeof



Rozbor testu → Úkol 2

```
function isObject(any) {  
  return Object(any) === any  
}
```

???

```
function isObject(any) {  
  return Object(any) === any  
}
```

```
function isObject(any) {  
  if (any === null) {  
    return false  
  }  
  if (typeof any === 'function') {  
    return true  
  }  
  if (typeof any === 'object') {  
    return true  
  }  
  return false  
}
```

```
function isObject(any) {  
  if (any === null) {  
    return false    typeof null === 'object'  
  }  
  if (typeof any === 'function') {  
    return true      typeof function(){} === 'function'  
  }  
  if (typeof any === 'object') {  
    return true  
  }  
  return false  
}
```


```
function isObject(any) {  
  if (any === null) {  
    return false  
  }  
  if (typeof any === 'function' || typeof any === 'object') {  
    return true  
  }  
  return false  
}
```



```
function isObject(any) {  
  if (!any) {  
    return false  
  }  
  if (typeof any === 'function' || typeof any === 'object') {  
    return true  
  }  
  return false  
}
```

```
function isObject(any) {  
  if (any && (typeof any === 'function' || typeof any === 'object')) {  
    return true  
  }  
  return false  
}
```

```
function isObject(any) {  
  return Boolean(any && (typeof any === 'function' || typeof any === 'object'))  
}
```



```
function isObject(any) {  
  return !!(any && (~['function', 'object'].indexOf(typeof any)))  
}
```

*Napište funkci `round(n, p)`
která vrátí číslo (`n`) zaokrouhlené
na určitý počet (`p`) desetinných míst.*

```
function round(n, p) {  
  return n.toFixed(p)  
}
```

```
typeof round(123.321, 2) // 'string' :(
```

```
function round(n, p) {  
  return n.toFixed(p)  
}  
  
typeof round(123.321, 2) // 'string' :(
```

```
function round(n, p) {  
    return Number(n.toFixed(p))  
}
```

```
function round(n, p) {  
    return +n.toFixed(p)  
}
```



```
function round(n, p) {  
  const multiplier = Math.pow(10, p)  
  return Math.round(n * multiplier) / multiplier  
}
```

// end