

**A 9SED Termékismertető – Számla és betéti bankkártya szkriptjeinek összeállítására vonatkozó követelmény-, valamint feltételrendszer**

2022. július 28.

Tartalom

[1. Bevezetés 3](#_Toc94176684)

[2. Képletek összeállításának és futtatásának általános szempontjai 4](#_Toc94176685)

[2.1. Sebesség 4](#_Toc94176686)

[2.2. Formai ajánlások 4](#_Toc94176687)

[2.3. Hibakezelés 4](#_Toc94176688)

[2.4. Futtatás 6](#_Toc94176689)

[2.5. Tesztelés és a Képletek feltöltése az ERA rendszerbe 6](#_Toc94176690)

[3. Képletekkel szembeni követelmények, elvárások 6](#_Toc94176691)

[3.1. Programozási nyelv, futtatási környezet 6](#_Toc94176692)

[3.2. Bemeneti paraméterek kezelése 7](#_Toc94176693)

[3.3. Kimeneti (számított) paraméterek kezelése 8](#_Toc94176694)

[3.4. Képlet kialakításával kapcsolatos követelmények 10](#_Toc94176695)

[4. Képletek összeállítására vonatkozó további korlátozások, tiltott kifejezések 11](#_Toc94176696)

[4.1. Korlátozások 11](#_Toc94176697)

[4.1.1. Tiltott kifejezések 11](#_Toc94176698)

[5. Képletek tesztelésének lehetősége az MNB infrastruktúrájához való hozzáférés nélkül 13](#_Toc94176699)

[5.1. Példa-tesztkörnyezet létrehozása 13](#_Toc94176700)

[5.2. Tesztelés, hibakeresés 13](#_Toc94176701)

[5.2.1. Használati javaslat 13](#_Toc94176702)

[6. Képletek feltöltésének módja az ERA rendszerbe 14](#_Toc94176703)

[7. Változások az előző verzióhoz képest 15](#_Toc94176704)

[8. Mellékletek 16](#_Toc94176705)

[1. számú melléklet: Be- és kimeneti paramétertábla 16](#_Toc94176706)

[2. számú melléklet: Input validáció 16](#_Toc94176707)

[3. számú melléklet: Output validáció 16](#_Toc94176708)

[4. számú melléklet: bemeneti és kimeneti mintaállományok 16](#_Toc94176709)

[5. számú melléklet: intézményi teszthez szimulációs keret és tesztszkript 16](#_Toc94176710)

[6. számú melléklet: szkriptajánló kiegészítés 16](#_Toc94176711)

# Bevezetés

Az adatszolgáltatóknak a …/2021. (… …) MNB rendeletben foglaltak szerint a forgalmazott és nem forgalmazott Számla- és betéti bankkártya termékeikre vonatkozó adatszolgáltatást az ERA rendszer „Számlatermékek” szolgáltatásán keresztül, JavaScript programozási nyelven összeállított szkriptként teljesíti.

Jelen dokumentum a JavaScript programozási nyelven rögzített szkript összeállításához, külső környezetben (vagyis: nem a Magyar Nemzeti Bank informatikai rendszerein) történő teszteléséhez és a szkript feltöltéséhez szükséges technológiai és logikai feltételeket tartalmazza. A dokumentum részletesen bemutatja továbbá a szkriptre vonatkozó követelményeket és megszorításokat, valamint az egyes bemeneti és kimeneti paraméterek használatának módját.

**Jelen feltételrendszer nem végleges: az az adatszolgáltatók által adott visszajelzések és a tesztelés során összegyűjtött tapasztalatok alapján az éles üzembe állásig változhat. A feltételrendszer módosításának jogát a Magyar Nemzeti Bank fenntartja.**

**Fogalomtár**

| **Fogalom** | **Magyarázat** |
| --- | --- |
| **ERA** | A Magyar Nemzeti Bank által üzemeltetett **e**lektronikus **r**endszer hitelesített **a**datok fogadásához.  |
| **Szkript**  | Az adatszolgáltató által összeállított, az MNB részére átadott, a „Számlatermékek” szolgáltatáson keresztül beküldött, JavaScript programozási nyelven összeállított adatszolgáltatás. A továbbiakban Képlet vagy komplex Képlet. |
| **Összehasonlító oldal** | Az MNB honlapján elérhető Bankszámlaválasztó program, valamint az MNB belső elemzési célú összehasonlítására szolgáló felület. |
| **Bemeneti paraméterek** | Azok a paraméterek, amelyeket a Képlet megkap az Összehasonlító oldaltól és amelyek alapján végrehajtja a Képletben rögzített számítást. |
| **Kimeneti paraméterek** | A Képlet-kiértékelő rendszer által a Képlet futtatását követően az Összehasonlító oldal számára visszaadott értékek. |
| **Képlet-kiértékelő rendszer/kiértékelő rendszer** | Az adatszolgáltató által összeállított, tesztelt és feltöltött Képleteket lefuttató rendszer.  |
| **Ajánlatadás** | Jelen dokumentumban „ajánlatadás” alatt azt kell érteni, hogy az adott számla/számlacsomag megfelel-e a felhasználó által megadott paramétereknek és így a termék az Összehasonlító oldal találati listájában megjelenik-e vagy sem.Az Összehasonlító oldalon keresztül szerződéskötés, ajánlatkérés nem kezdeményezhető, a felület kizárólag tájékoztatásra szolgál. Az Összehasonlító oldal keretében nyújtott tájékoztatás nem minősül a Polgári Törvénykönyv szerinti ajánlattételnek, az itt szereplő adatok kizárólag tájékoztató jellegűek. |

# Képletek összeállításának és futtatásának általános szempontjai

## Sebesség

1. A Képleteket úgy kell összeállítani, hogy azok kiértékelése a rendszerben a lehető leggyorsabban és hibamentesen megtörténhessen. Ezért alapvető elvárás a Képletek optimalizálása és a felesleges utasítások elhagyása[[1]](#footnote-2). Figyelemmel kell lenni az általános programozói és informatikai elvek betartására[[2]](#footnote-3), illetve – lehetőség szerint – kerekítések alkalmazására.
2. A futtatási idő minimalizálása érdekében kifejezetten javasolt a Képletek minifikálása vagy obfuszkálása, amennyiben a megfelelő működés továbbra is biztosított.
3. A Képletekben minden olyan matematikaifüggvényhasználható – és kifejezetten javasolt – amelyet a futtató környezet[[3]](#footnote-4) támogat.
4. A végtelen ciklusok okozta számításikapacitás-szükséglet minimalizálása érdekében a kiértékelő rendszer a Képlet futását 600 ezredmásodperc után hibajelzés mellett automatikusan leállítja.
5. Az ERA rendszerben feltöltendő JavaScript Képletek mérete korlátozott. A rögzítendő Képletek .txt fájlként tárolt állományának mérete nem haladhatja meg – termékenként – az 1 Mbyte méretet.
6. A Képletekből el kell távolítani az összeállítás során rögzített megjegyzéseket, nem futó (futtatott) programkód részleteket[[4]](#footnote-5), automatikus eszközök által generált kiegészítő információkat.
7. A fenti kritériumoknak való megfelelés esetén a Képletekben megfeleltetési táblázatok (pl.: többdimenziós tömb, object-map), előre elkészített mátrixok is használhatók.
8. A Képletek futtatásához szükséges memóriaigény – termékenként – nem haladhatja meg az 4 Mbyte méretet.
9. A scriptnek tartalmaznia kell a működéséhez szükséges összes adatot (kivéve az input adatokat). Más forrásból származó adatok elérésére nincs lehetőség.

*A Magyar Nemzeti Bank fenntartja a jogot arra, hogy felfüggessze a termék Összehasonlító oldalon történő megjelenítését, amennyiben a termékhez feltöltött Képlet rendszeresen hibát okoz, vagy a Képlet futását a kiértékelő rendszer rendszeresen leállítja.*

## Formai ajánlások

1. A Képletek kizárólag az alap ASCII karakterkészletet tartalmazhatják, így különösen nem használhatók magyar ékezetes és egyéb speciális karakterek. (E szabály alól természetesen kivételek azon szövegek, amelyek a szkriptben szereplő paraméterek értékeiként értelmezendők, pl. szöveges leírások.)
2. A Képletek az ERA rendszerbe történő feltöltéskor vírusellenőrzésen esnek át, ezért minden olyan kifejezés használatát mellőzni kell, amelyet a vírusellenőrző gyanúsnak ítélhet meg.
3. Az ERA és a Képletkiértékelő rendszer a Képletek működését szemantikai, illetve üzleti szempontból nem elemzi. A rendszer ugyanakkor a Képletek ERA rendszerbe való betöltését megelőzően szűri a tiltott kifejezéseket.
4. A JavaScript kisbetű-nagybetű-érzékeny programozási nyelv, ezért kiemelten fontos a bementi és kimeneti paraméterek helyes használata! A kiértékelő rendszer által biztosított paraméterek megnevezései kizárólag kisbetűkből, aláhúzás jelből és számokból állnak.

## Hibakezelés

1. **A hibák kezelése az adatszolgáltató feladata**, a Képletkiértékelő rendszer a hibás Képletek javítására, illetve ismételt meghívására nem tesz kísérletet, egy ajánlatadáson belül minden Képlet meghívására csak egyszer kerül sor.
2. A bemeneti paraméterek adott értékei mellett a Képletben esetleg hibát okozó kifejezések kezelése az adatszolgáltató feladata (pl.: 0 értékű operandussal történő osztás).
3. A rendszer biztosítja, hogy minden kötelező paraméternek legyen kezdeti bemeneti értéke. Az egyéb paraméterek tekintetében az alábbi, 6. pontban és az 1. számú mellékletben található információk a mérvadók.
4. Az MNB nem felel azon jogkövetkezményekért, kárért, amely annak következtében áll elő, hogy az adatszolgáltató szándékától eltérően, vagy egyéb okból hibásan töltötte fel a Képletet.
5. Az adatszolgáltató által készített Képlet többféle adatforrásból kaphat bemeneti adatokat, függően attól, hogy az MNB összehasonlító oldala részeként vagy ellenőrzés, illetve tesztelés céljából használják. Ezért kiemelten fontos, hogy az adatszolgáltató a lehető leghibatűrőbb megoldásokat alkalmazza a Képlet megírásakor. Ez különösen érvényes a bemeneti és kimeneti adatok ellenőrzésére.
6. **Az MNB kötelezően előírja a bemeneti adatok (paraméterek) használata előtti ellenőrzést**, amelynek ki kell terjednie az adott paraméter **létezésének és értelmezhető értékének** ellenőrzésére. Az ehhez használt egyik függvény*minta* az alábbi:



1. Az esetlegesen üres stringként érkező bemeneti paraméterek számmá konvertálása (amennyiben az adott paraméter elvárt típusa/formátuma numerikus) szintén kötelező, mivel a JavaScript nyelv nem szigorúan típusos, és az üres stringgel végzett számtani műveletek (pl. összeadás) az eredmény típusát módosítják, ami a számítások esetén komoly eltéréseket okozhat. Pl.:

var inputParam = ’’ [üres string] esetén:

100 + inputParam -> 100,

100 - inputParam -> 100,

100 \* inputParam -> 0,

100 / inputParam -> Infinity

A fenti esetekben a végeredmény már nem szám, hanem szöveg típusú lesz.

Az alábbi példa megmutatja, hogy az üres string felhasználása a kalkulációban milyen veszélyekkel járhat:

100 + inputParam + 10 = 10010

Ezért az üres stringként kapott értékeket tilos a kalkulációban típuskonverzió és egyéb ellenőrzések nélkül felhasználni.

Az üres stringek és NULL értékek numerikus értékre (0-ra) való konverzióját az alábbi példa szerint lehet elvégezni:

var x = Number(inputparam);

## Futtatás

1. A rendszer a Képlet által biztosított kimeneti paramétereket nem vizsgálja, nem elemzi. **Az adatszolgáltató feladata, hogy a kimeneti paramétereket a kimeneti validációnak megfelelően állítsa be, függetlenül attól, hogy ad-e végül érvényes ajánlatot vagy sem.**
2. A Képletek szemantikailag hibás futásából származó (pl.: negatív értékű bankszámlahasználat becsült költségének eredménye) kockázatok csökkentése érdekében javasoljuk a kimeneti paraméterek ellenőrzését a rendszernek történő eredmények visszaadását megelőzően. A kimeneti paraméterekkel kapcsolatos részletes információkat a 3.3. alfejezet tartalmazza.
3. Az adatszolgáltatónak lehetősége van arra, hogy a fogyasztó által megadott bemeneti paraméterek alapján ne adjon ajánlatot. **Ebben az esetben az „ajanlatadas” mezőt FALSE értékre, az összes többi kimeneti paraméter értékét NULL értékre kell állítani**. Az optimális futásidő érdekében a bemeneti paraméterek vizsgálatát a Képlet futásának elején kell végrehajtani. Amennyiben egy kimeneti paraméter a Képlet egy korábbi szakaszában kapott valid értéket, azonban a Képlet egy későbbi szakaszában az adatszolgáltató úgy dönt, hogy nem szándékozik ajánlatot adni, akkor a kimeneti paraméterek közül az „ajanlatadas” mezőt FALSE értékre, az összes többi kimeneti paraméter értékét NULL értékre kell állítania.
4. A Képlet-kiértékelő rendszerben használt Java futtatókörnyezet verziófrissítését megelőzően a Magyar Nemzeti Bank előzetes tájékoztatást ad az adatszolgáltató számára, felkészülési időt biztosítva a frissítéssel járó esetleges módosítások átvezetésére.
5. A Képlet-kiértékelő rendszer terhelése számos tényezőtől függ, ezért a rendszer nem garantálja, hogy ugyanazon Képlet ugyanolyan paraméterekkel minden esetben pontosan ugyanannyi idő alatt fut le.

## Tesztelés és a Képletek feltöltése az ERA rendszerbe

1. A Képleteket az ERA rendszerbe történő feltöltést megelőzően javasolt minél több bemenetiparaméter-kombinációra tesztelni. Az ERA rendszerben történő tesztelés a végső ellenőrzést szolgálja. A Képletek MNB rendszerén kívül történő tesztelésének lehetőségéről az 5. fejezet ad áttekintést.
2. Felhívjuk a Hitelintézetek figyelmét, hogy az ERA rendszerbe feltöltött Képletek NEM jelennek meg azonnal az Összehasonlító oldalon, csak az ütemezett publikációs eljárást követően. Ezért a Képlet Összehasonlító oldalon való megjelenéséig a naponta egyszer történő publikáció miatt a feltöltéstől számítva akár 24 óra is eltelhet.
3. Az MNB által megjelölt kapcsolattartók és az ERA honlapon feltüntetett Helpdesk az ERA rendszer használatával kapcsolatos általános és technikai jellegű problémákkal kapcsolatban biztosítanak támogatást. A Képletek összeállításában, valamint azok szemantikai ellenőrzésére vonatkozóan az MNB nem biztosít szolgáltatást az adatszolgáltatók számára. A Képletek megfelelő összeállítása az adatszolgáltatók feladata és felelőssége.

# Képletekkel szembeni követelmények, elvárások

## Programozási nyelv, futtatási környezet

A Képletek programozási nyelvére vonatkozó követelmények:

1. A Képletek összeállítása kizárólag JavaScript programozási nyelven történhet – egyéb programozási nyelvek nem támogatottak.
2. Minden olyan JavaScript feltölthető, amelyet a kiértékelő rendszer képes lefuttatni, és megfelel az alábbi feltételeknek:
	1. JavaScriptek futtatása strict módban történik: a „use strict” direktívát a rendszer automatikusan bekapcsolja a Képlet kiértékelése előtt;
	2. a Nashorn Engine le tudja futtatni a Képletet, és az adatszolgáltató által elvárt eredményt adja vissza.
3. A Képletek megfelelnek az ECMAScript 5.1 specifikációnak.

A Képletek futtatása az alábbi futtatási környezetben történik:

1. JAVA8 (Oracle) JRE: 1.8.0\_231 64 bites Java futtató környezet Windows Server 2019 operációs rendszerre telepítve;
2. A fenti JAVA futtatókörnyezet Nashorn JavaScript engine-jét felhasználva.

A Nashorn-nal kapcsolatban további információk találhatóak többek között például az alábbi URL elérhetőségeken:

<http://www.oracle.com/technetwork/articles/java/jf14-nashorn-2126515.html>

<http://winterbe.com/posts/2014/04/05/java8-nashorn-tutorial/>

<http://www.javaworld.com/article/2144908/scripting-jvm-languages/nashorn--javascript-made-great-in-java-8.html>

<http://www.n-k.de/riding-the-nashorn/>

## Bemeneti paraméterek kezelése

**Bemeneti paraméterek értékadása**

A bemeneti paraméterek az Összehasonlító oldalon a fogyasztó által bevitt adatok alapján kapják meg értéküket. Az ERA oldalon pedig a feltölthető JSON-tesztadatállomány megfelelő mezőibe írhatók a megfelelő értékek.

**Bemeneti paraméterek elnevezése**

A jelen dokumentum 1. számú mellékleteként kiadott paramétertáblázat tartalmazza, hogy az Összehasonlító oldalon adott címkével jelölt beviteli mező értékére milyen paraméternévvel lehet hivatkozni a Képletben. Ismételten felhívjuk a figyelmet arra, hogy a Képlet-kiértékelő rendszer által biztosított paraméterek megnevezései kizárólag kisbetűkből, aláhúzás jelből és számokból állhatnak!

Pl.:

|  |  |
| --- | --- |
| **Paraméter címkéje az Összehasonlító oldalon** | **Paraméter Képletben hivatkozható neve** |
| Számla típusa | szamla\_tipus |
| Mobilbanki szolgáltatás | mobilbank |

Az Összehasonlító oldal a Képletek meghívása előtt validálja a fogyasztó által megadott értékeket, így a Képletek csak érvényes[[5]](#footnote-6) bementiparaméter-értékekkel futnak le.

**Ismétlőblokk kezelése**

A beviteli paraméterek esetében ismétlőblokknak azt a blokkot nevezzük, amelyekből a felhasználó az összehasonlító oldalon akár többet is megadhat, és egy adott paramétercsoportot fog össze. Ilyen pl. a kártyatípusokra vonatkozó paraméterek blokkja. A kártyatípusok adatait szabványos JSON array-ként adja át a rendszer a Képlet számára.

Az ismétlőblokkok száma limitált, az egyes mezőcsoportoknál előforduló ismétlőblokkok maximális számát az 1. számú mellékletben található leírás tartalmazza.

A 4. számú mellékletben található I/O mintaállomány tartalmazza az ismétlőblokkok megadásának helyes módját is.

**Értékkészletek (kiválasztó mezők) kezelése**

Az értékkészlet típusú bemeneti paraméterek esetében a rendszer szintén megfeleltetést alkalmaz az Összehasonlító oldalon található szöveges címkék és a Képletekben használható értékek között.

Pl.:

|  |  |
| --- | --- |
| **Érték címkéje** | **Képletben hivatkozható érték** |
| SMS-szolgáltatás típusa | sms |
| Kivonat típusa | kivonat |

**Bemenetiparaméter-értékek kiolvasása a Képletben**

A Képlet a bemeneti paramétereket egy object mapben kapja meg, amelyen belül minden paraméter értékéhez a nevére való hivatkozással lehet hozzáférni. Az object map neve: ”inputparams”

Az adott bemeneti paraméter értékéhez az inputparams.valtozo\_nev –en keresztül lehet hozzáférni.

***Példák****:*

1. A „Számla típusa” változó értékének beolvasása: a lekérdező oldalon „Számla típusa” az 1. számú melléklet alapján az ehhez tartozó paraméter neve: „szamla\_tipus”.

Így az adott paraméter értékét pl.

inputparams.szamla\_tipus;

kifejezéssel lehet megkapni, mely eredményül pl. ezt adja: csomag\_alapu

2. Az előző paraméter érték saját változóban való eltárolása pl.:

var szamla\_tipus = inputparams.szamla\_tipus;

3. A bemeneti paraméterek értékeihez a fentieken túl a szabványos JavaScript nyelv által támogatott inputparams[’valtozo\_nev’] módon is hozzá lehet férni.

Például: inputparams[’szamla\_tipus’]

**Bemeneti paraméterek által visszaadott értékek**

A bemeneti paraméterek az alábbi értékeket vehetik fel:

1. numerikus (szám) típusú mezők esetében: diszkrét szám, érték, pénzegység, illetve egyéb formátumjelölő **nélkül**;
2. nem megadott (opcionális) paraméter esetén a paraméter értéke üres (NULL);
3. értékkészlet típusú mezők esetében vagy üres (NULL), vagy pedig az adott címkéhez tartozó megfelelő érték;
4. kiválasztómezők (checkbox) esetében vagy üres (NULL), vagy pedig az 1. számú mellékletben található érték.

## Kimeneti (számított) paraméterek kezelése

**Amennyiben az adatszolgáltató ad ajánlatot, akkor az összes kimeneti paraméternek numerikus, logikai (igaz/hamis), dátum, NULL vagy szöveg típusú értéket kell felvennie a Képlet kiértékelése során.** Amennyiben a Képlet alapján bármely kimeneti paraméter nem vesz fel értéket („undefined”), akkor a kimeneti paraméterek validációja hibát jelez, és a Képlet nem alkalmas az MNB rendszerén belüli működésre.

Ha valamely kimeneti paraméter minden gondosság és tesztelés ellenére, már az éles működés során nem vesz fel értéket, azt a kiértékelő rendszer úgy értékeli, hogy az adatszolgáltató nem adott ajánlatot az Összehasonlító oldalon megadott bementi paraméterek alapján. Ilyen esetben az Összehasonlító oldal a terméket nem jeleníti meg.

**Kimeneti paraméterek elnevezései és kerekítési határai**

A bemeneti és **kimeneti paraméterek teljes specifikációját az 1. számú melléklet tartalmazza**.

1. **Mindegyik kimeneti paraméterre igaz, hogy csak nulla vagy annál nagyobb (pozitív) számértéket, szöveget, dátumot és NULL értéket lehet beállítani eredménynek. Ha a szkript a bemeneti paraméterek alapján nem tud érvényes ajánlatot adni, vagy bármely okból hibára futott, akkor az „ajanlatadas” mező értékét FALSE-ra kell állítani, és minden egyéb kimeneti paraméter értékét NULL-ra.**
2. Ajánlatadás esetén a kimeneti paraméterek elvárt típusa „number”, „boolean”, „null”, „date” és „string”. Más típus (pl. Object() stb.) vagy nem létező (nem inicializált) paraméter („undefined”) hibás kimeneti validációt eredményez. **A nem elvárt típusú kimeneti érték(ek)et visszaadó szkript nem alkalmas az MNB rendszerén belüli működésre.**
3. **Abban az esetben, ha nem adható érvényes ajánlat, az „ajanlatadas” mező értékét FALSE-ra kell állítani, és minden egyéb kimeneti paraméter értékét NULL-lal kell feltölteni.**
4. A kimeneti paraméterek értékét a rendszer a JavaScriptben alkalmazott adattípustól függetlenül kerekíti az adott kerekítési határnál. A kerekítés az általános kerekítési szabályok szerint történik, vagyis nullától négyig lefelé, afölött pedig felfelé. Például a bankszámlahasználat becsült költségénekértéke a kimeneti változóban 10000.67, akkor a rendszer a felületen a 10001 értéket jeleníti meg. Felhívjuk a figyelmet arra, hogy a kimeneti paraméterek értékét az előbb leírt kerekítési szabály figyelembe vételével kell megadni[[6]](#footnote-7)!
5. Az ECMAScript specifikáció alapján a számok kezelése dupla precizitású 64-bites bináris formátumú IEEE 754 érték. Kiemelten felhívjuk a Hitelintézet figyelmét az ebből adódó számítási pontosság figyelembe vételére! Például 0.1 + 0.2 = 0.30000000000000004, amelynek kerekítését a kiértékelő rendszer a JavaScript környezeten kívül végzi BCD kódolással.
6. A numerikus kimeneti paraméterekben szöveg típusú értéket (például e, ezer, %, százalék, száz, Ft, forint, HUF stb.) megjelölni szigorúan tilos!
7. Kimeneti paraméternek függvényhivatkozást, dinamikusan számított propertyt (Object.defineProperty) beállítani szigorúan tilos! Kimeneti paraméternek kizárólag konkrét eredmény értékét, illetve NULL-t lehet beállítani.
8. A kimeneti paramétereket részeredmények, ideiglenes eredmények tárolására használni szigorúan tilos!

*A Képletek szemantikailag hibás futásából származó (pl.: negatív értékű* bankszámlahasználat becsült költsége*) kockázatok csökkentése érdekében javasoljuk a kimeneti paraméterek alapos és szisztematikus ellenőrzését a kiértékelő rendszerbe való feltöltést megelőzően! Az Összehasonlító oldal kizárólag abban az esetben jeleníti meg a számlaterméket, amennyiben az összes kimeneti paraméter valid értéket vesz fel.*

**Kimenetiparaméter-értékadás**

A kimeneti paramétereknek az „output” objektum „values” adattagjának object map-jén keresztül lehetséges értéket adni. Ebből adódóan a kimeneti értékadás: output.values.kimeneti\_parameter\_nev=érték

Például:

output.values.eletkor\_minimum=18

A rendszer a kimeneti paraméterek értékei közötti összefüggéseket részben vizsgálja (output validáció). Amennyiben az adatszolgáltató figyelembe vesz bizonyos határértékeket az ajánlatadásra vonatkozóan és az az input paraméterek alapján megállapítható, akkor azt kérjük a Képletben lekezelni (pl.: a felhasználó által megadott életkor, igénybe venni kívánt szolgáltatás alapján).

## Képlet kialakításával kapcsolatos követelmények

**Állapotmentes működés**

1. Alapvető követelmény a Képletekkel kapcsolatosan, hogy azok állapotmentesen működjenek, vagyis két azonos paraméterhalmazzal történő behívás esetén pontosan ugyanazt az eredményt adják. **A Képletek működése nem függhet a futtatás időpontjától**, előző Képletfuttatás eredményétől, illetve más, külső vagy véletlen tényezőtől. A rendszer automatikusan kezeli az ERA-ban a termékfeltöltő adatlapon beállított publikálási vagy érvényességi időpontot.
2. Tájékoztatjuk az adatszolgáltatót, hogy a termékek publikációja az ERA beküldő rendszerből az Összehasonlító oldalra naponta egy alkalommal történik.

**Sebesség**

A Képletek az Összehasonlító oldal működését hivatottak biztosítani, ezért elvárás, hogy a Képletek az MNB rendszerében maximum 600 msec (600 ezredmásodperc) alatt lefussanak. A futási idő tekintetében az ERA rendszer kijelölt üzemidejében (munkanapokon reggel 8 és 20 óra között) a Képletfeltöltő felületén a Képlet tesztelésekor (lásd később) megjelenő futtatási időt szükséges alapul venni.

**Képlet meghívásához szükséges belépő függvény kialakítása**

Az MNB rendszerébe kizárólag a következőkben meghatározott metódus szignatúrával lehet a Képletet feltölteni (egyéb esetben a rendszer a Képletet nem fogja meghívni). A függvény neve és paraméterei rögzítettek, azokat megváltoztatni szigorúan tilos! A belépő függvényen belül lehetőség van saját változók, további függvények definiálására.

A belépő függvény neve: calculate (kis-nagybetű érzékenyen, pontosan ebben a formában)

Paraméterei:

1. inputparams (ebben adja át a rendszer a bemeneti paramétereket)
2. output (ebben adja át a Képlet a kimenő paraméterek értékeit)

A függvényt pontosan ebben a formában kell deklarálni:



Példák a belépő függvény kialakítására:



Az előző példa kiegészítve saját szorzó függvény használatával:



**Obfuszkálás**

A feltöltésre kerülő Képlet méretének és futtatási idejének minimalizálása érdekében kifejezetten javasolt a Képletek minifikálása vagy obfuszkálása, amennyiben a megfelelő működés továbbra is biztosított. Az obfuszkálást úgy szükséges elvégezni, hogy a rendszer által kezelt belépő függvény szignatúráját; bemeneti és kimeneti paraméterek neveit ne változtassa meg (pl.: az obfuszkáláskor kivételként rögzíteni), vagy valamely megfeleltetési módszerrel biztosítani a paraméterek működőképességét

(pl.: nptprms.szml\_tps= inputparams.szamla\_tipus).

A Képleten belül van lehetőség saját függvények definiálására, így az üzletileg kritikus kódrészek akár külön függvénybe is szervezhetőek és csak az adott függvény szintjén obfuszkálhatók.

# Képletek összeállítására vonatkozó további korlátozások, tiltott kifejezések

A „Képletek összeállításának és futtatásának általános szempontjai” fejezetben leírtakon túl a Képletekre az alábbi korlátozások vonatkoznak. A tiltott kifejezést tartalmazó Képlet futtatását a rendszer megtilthatja/hibát okozhat az Összehasonlító oldalon.

*A Képletkiértékelő rendszer biztonsági okokból alapértelmezés szerint tiltó szabályrendszert alkalmaz, vagyis külső programkönyvtárak, Java kódrészletek a Képletekből nem hívhatók meg. A Képletekben továbbá nem használhatóak olyan kifejezések, amelyek a globális névtérhez való hozzáférést lehetővé tennék (pl.: global, Function, eval, stb…), illetve aszinkron folyamatot indítanak (pl.: setTimeout, setInterval, setImmediate). Szigorúan tilos bármilyen szándékosan kártékony célú Képletet a rendszerbe feltölteni! Az adatszolgáltató az általa feltöltött Képletekért felelősséget vállal, és a szándékos károkozás jogkövetkezménnyel jár.*

## Korlátozások

1. A JavaScriptből Java függvényeket, kifejezéseket meghívni tilos!
2. A Nashorn által alapértelmezés szerint biztosított Java-s kiterjesztések nem használhatók!

### Tiltott kifejezések

1. load() függvény használata;
2. eval() használata;
3. Function() használata;
4. setTimeout() használata;
5. setInterval() használata;
6. setImmediate() használata;
7. XMLHttpRequest() használata;
8. DOM manipulációra vonatkozó kifejezések, pl: document;
9. Naplózási kifejezések: console;
10. Kilépést lehetővé tevő, illetve egyéb, nem a számításhoz szükséges függvények:
	1. quit()
	2. exit()
	3. print()
	4. echo()
	5. readLine()
	6. readFully()
	7. loadWithNewGlobal()
	8. bindProperties
11. globális objektumok elérése:
12. $ARG
13. $ENV
14. $EXEC()
15. $OUT
16. $ERR
17. $EXIT
18. A Képletekben hálózati kapcsolatot igénylő funkciók használata szigorúan tilos! Ez vonatkozik az interneten elérhető JavaScript modulok, könyvtárak használatára is. A Képletnek mindenképpen szinkronosan kell az eredményt visszaadnia, aszinkron kódfuttatásra nincs lehetőség.

# Képletek tesztelésének lehetősége az MNB infrastruktúrájához való hozzáférés nélkül

Javasolt a Képleteket az MNB ERA rendszerébe történő feltöltést megelőzően körültekintően tesztelni mind a működés, mind a futási idő tekintetében. A következőkben javaslatot teszünk a Képletek külső környezetben történő tesztelésére a Magyar Nemzeti Bank környezetének szimulálásával. Az így elvégzett sikeres tesztet követően várhatóan az MNB rendszerében is megfelelően lefut a megadott Képlet.

*Figyelem! A tesztkörnyezet létrehozásával kapcsolatos javaslatok alapvetően tájékoztató jellegűek. Javasoljuk mindezek mellett összetettebb és bővebb tesztelési lehetőséget biztosító környezetek alkalmazását is.*

## Példa-tesztkörnyezet létrehozása

A tesztkörnyezet létrehozásához szükséges lépések (Windows operációs rendszer esetén. Linux és egyéb operációs rendszerek esetén a lépések hasonlóak az itt leírtakhoz az adott operációs rendszerrel kapcsolatos eltéréseket értelemszerűen figyelembe véve):

1. (Oracle) JAVA8 futtató környezet telepítése az adott számítógépre.
2. A Nashorn rendelkezik egy parancssori futtató klienssel, amely a feltelepített JAVA bin alkönyvtárából érhető el, jjs.exe fájlnévvel.
3. A tesztelés megkönnyítése érdekében célszerű a jjs.exe helyét felvenni a „PATH” környezeti változóhoz, így a rendszerből bárhonnan egyszerűen meghívható.
4. Jelen dokumentum 5. számú mellékleteként átadott MNB\_bszv\_keplet\_ajanlas\_keret.js fájl egy olyan JavaScript keretet ad, ami szimulálja az MNB-s rendszer működését. Ehhez bizonyos beállításokat elvégez, illetve inicializálja a bemeneti paramétereket (mintha az Összehasonlító oldalról érkeznének az értékeik). Az 5. számú mellékletben átadott fájlt módosítani szükséges olyan módon, hogy az abban megjelölt helyre kerüljön beillesztésre az adatszolgáltató által az ERA rendszerbe feltölteni kívánt JavaScript kód. A beillesztésre kerülő kódnak pontosan ugyanannak kell lennie, mint amit az ERA rendszeren keresztül is fel fognak tölteni.
5. Az ajánlás 5. számú mellékleteként átadott példa JavaScript csak néhány kötelező bemeneti paraméterek bemenő adataira tartalmaz példaértékeket, és néhány kimenő paraméter értékét kiírja a konzolra. Amennyiben más bemeneti értékek tesztelése a cél, akkor célszerű a scriptet ennek megfelelően értelemszerűen módosítani, esetleg átírni olyan módon, hogy a paraméterek értékeit parancssori argumentumból vegye, és az operációs rendszerben futtatható scriptet írni a megfelelő paramétert értékek meghívásával. Figyelem! Az ERA rendszerbe feltöltendő Képlettel kapcsolatos elvárásokat célszerű ebben az esetben is szem előtt tartani!
6. **Fontos! A kötelező (required) kimeneti paramétereket minden esetben be kell állítani a számítások elvégeztével!**
7. Ugyanabban a mappában állva, ahol az MNB\_bszv\_keplet\_ajanlas\_keret.js fájl is található, parancssorból pl. az alábbi utasítással indítható a példa script futtatása:

jjs MNB\_bszv\_keplet\_ajanlas\_keret.js

## Tesztelés, hibakeresés

A kimeneti paraméterek listája kiegészült egy opcionális, string típusú paraméterrel: „errorMessage”.

### Használati javaslat

Az „errorMessage” megjelenik a kimeneten (a szkript feltöltéséhez kapcsolódó tesztfuttatáskor), de a felhasználók által látott kalkulátorban nem, ezért tetszőleges (string típusú vagy stringre konvertált) értéket kaphat, hibakeresés céljából.

* Pl. beleírható egy komplex részkalkuláció eredménye, ami így külön is tesztelhetővé válik.
* A másik lehetőség, hogy az ajánlatadás elmaradását lehet figyelni általa: azon a ponton, ahol (pl. egy részszámítás miatt) az ajánlatadás false-ra fut, az errorMessage-be beírható az eseménnyel kapcsolatos megjegyzés (pl. „túl alacsony havi jóváírás” stb.).
* Ha a szkript az MNB által javasolt try-catch-blokkot használja keretként, akkor a catch-ágban megjelenő error-t (catch(error)) {}) bele lehet írni, így a catch-ágra futás oka jelenik meg az errorMessage-ben.
* A catch-ágra futtatást saját üzenetekkel is elő lehet idézni, ilyenkor a saját hibaüzenet jelenhet meg az errorMessage-ben. Pl. throw ”Túl alacsony havi jóváírás: ” + havi\_jovairas\_osszege;

# Képletek feltöltésének módja az ERA rendszerbe

Az elkészített és megfelelően tesztelt Képleteket az ERA rendszeren keresztül lehet beküldeni az MNB rendszerébe.

*Kizárólag megfelelően tesztelt és hibamentes Képleteket lehet feltölteni az MNB rendszerébe!*

1. Az ERA rendszerbe megfelelő jogosultsággal rendelkező felhasználóval belépve, majd a megfelelő szolgáltatást kiválasztva megjelenik a korábban rögzített termékeket listázó felület, valamint az „Új termék” funkció gomb.
2. Egy új vagy már meglevő termék adatlapját megnyitva a „Számolt értékek” mezőnél a „Script feltöltése” gombra kattintva megjelenik a Jogi és felelősségvállalási nyilatkozat, melyet elfogadva jelenik meg a Képlet feltöltő felület, valamint a teszt JSON fájl feltöltésére szolgáló felület.



1. A fájlok feltöltését követően a „Tesztelés” gombot megnyomva a rendszer a fájlokon vírus ellenőrzést végez, majd lefuttatja azt, és megjeleníti a Képlet által visszaadott eredményeket táblázatos formában. Minden egyes példa paraméterhez tartozó eredmény külön sorban jelenik meg (a példában minden eredmény értéke 0), az utolsó oszlopban pedig a rendszer által mért futási idő található msec-ban. Az eredményeket a listázáskor a rendszer nem egészíti ki egyéb jelölőkkel (pl.: % jel, Ft, stb.).
2. Az MNB a Képletek futási idejét az ERA rendszerben a Képlet tesztelésénél kiírt futási idők alapján határozza meg. A Képletek futási idejének nem szabad elérniük a 600 ezredmásodpercet. Az ERA a futási időbe nem számítja bele a víruskeresés, illetve a script ellenőrzésének idejét.
3. Az ERA által kijelzett futási idő az ERA terhelésének függvényében adott időpontokban eltérő lehet. Az ERA csak az adott példaparaméterekkel futtatott kiértékelés futási idejét jeleníti meg, ez azonban nem jelenti azt, hogy ugyanazon Képlet más paraméter értékékekkel szintén ugyanennyi idő alatt futna le (például: másik feltételes ágon több számítási művelet történik).
4. A Képlet csak akkor menthető le, ha a JavaScript sikeresen lefuttatható. Hibás (nem futtatható) vagy vírusos JavaScript nem menthető.
5. A Képlet hagyhat üresen kimeneti paramétereket, az output validációban meghatározott szabályok szerint, ez a Képlet mentését nem akadályozza meg. Az output validáció szabályait a 3. számú melléklet tartalmazza.
6. A Képlet mentését követően az adott termék szerkesztő adatlapját is szükséges menteni.

*Figyelem! Az Összehasonlító oldal csak az ERA rendszerbe beküldött, és az ERA felületén publikált állapotú termékeket jeleníti meg. Az ideiglenesen mentett termékek nem jelennek meg az Összehasonlító oldalon.*

# Változások az előző verzióhoz képest

* Az 1. sz. mellékletben található paramétertábla technikai pontosítást tartalmaz (az „output értékkészlet” munkalapon a módosult cella kékkel került kiemelésre).
* A 2. számú mellékletben található input validáció nem tartalmaz változást.
* A 3. számú mellékletben található output validációban pontosítás történt a Betétikártya gyártási/kibocsátási díja paraméter vonatkozásában.
* A 4. számú melléklet bemeneti mintaállományában technikai pontosítási történt.
* Az 5. számú melléklet nem tartalmaz változást.
* A 6. számú mellékletben található szkirptajánló kiegészítés – korrektúrával – technikai pontosításokat tartalmaz, valamint kiegészítésre került a „Foglalkozás, státsusz” input paraméterrel..

# Mellékletek

## számú melléklet: Be- és kimeneti paramétertábla



## számú melléklet: Input validáció

**

## számú melléklet: Output validáció



## számú melléklet: bemeneti és kimeneti mintaállományok



## számú melléklet: intézményi teszthez szimulációs keret és tesztszkript



## számú melléklet: szkriptajánló kiegészítés



1. Például csak hibakeresési célra szolgáló mellékszámítások, naplózások stb. beépítése. [↑](#footnote-ref-2)
2. Például egy eredmény a „for” ciklus közepén már ismert, akkor a „for” ciklus további futtatására már ne kerüljön sor; optimális keresési, közelítő eljárások alkalmazása stb. [↑](#footnote-ref-3)
3. A futtató környezettel kapcsolatos követelményeket részletesen lásd a 3. fejezetben. [↑](#footnote-ref-4)
4. Például a „kikommentezett” függvényeket. [↑](#footnote-ref-5)
5. A bementi paraméterek validációs szabályait a 2. számú melléklet tartalmazza. [↑](#footnote-ref-6)
6. Egy példával élve: amennyiben egy kimeneti paraméter értékét 3.4567 értékben célozza meg az adatszolgáltató, a Képletben ezt 3.45 értékben kell meghatározni. Az Összehasonlító oldalon ugyanis automatikusan – a kerekítési szabályok miatt – 3.46 érték jelenne meg. [↑](#footnote-ref-7)