

DEBRECENI EGYETEM

Informatikai Kar

**Üzleti intelligencia a gyakorlatban
feladatgyűjtemény**

EFOP-3.4.3-16-2016-00021 projekt támogatásával készült

Sikolya-Kertész Kinga

**Debrecen
2017**

1. SAS Enterprise Guide használatának alapjai

A feladatgyűjteményben használt állományok az Orion Star Sports & Outdoors kisvállalat fiktív adatai (letölhetőek: <https://arato.inf.unideb.hu/sikolya.kinga/files/SAS%20mintafajlok>).

1. Készítsünk projektet az *CUSTOMERS* adatállományra! Térképezzük fel az állomány változóit!
2. Készítsünk projektet az *EMPLOYEE_ORGANIZATION* adatállományra!
 - a) Térképezzük fel az állomány változóit!
 - b) Az adathalmazról kérjük le az alapinformációkat!
3. Hozzunk létre egy könyvtárhivatkozást (Name: *ORION*) az Orion Star Sports & Outdoors cég adatállományaira!
4. Készítsünk másolatot a *TRAVEL_EXPENSES* adattábláról a Work könyvtárba! A műveletet a szerver listában hajtsuk végre!
5. Hajtsuk végre a kért műveleteket a *EMPLOYEE_MASTER* táblán!
 - a) Keressük meg a Find parancs segítségével a Warehouse Manager beosztást a Job_Title oszlopban!
 - b) Rögzítsük le az Employee_ID változót!
 - c) Rejtsük el a Salary és a Birth_Date változókat!
 - d) Oldjuk fel az Employee_ID változót és állítsuk vissza a kiinduló táblát!
 - e) Helyezzük át a Manager_ID változót az Employee_Name jobb oldalára!

2. Adatok importálása

1. Importáljuk be az EG-be a *PRODUCTS* Excel fájl ProductList tábláját a következő utasításokat figyelembe véve
 - a) az első oszlop elnevezése legyen Product_ID a címkézése pedig Product ID;
 - b) a Supplier_ID típusát módosítsuk karakteresre ("String");
 - c) az Import Data Task az Import Products nevet kapja!

2. Importáljuk be az EG-be a *ORDERS.TXT* szöveg fájlt a változók tulajdonságainak alábbi módosításait figyelembe véve
 - a) hagyjuk ki az *Employee_ID* változót a táblából;
 - b) a *Discount* változó kimeneti formátuma legyen *PERCENTw.d* (5 szélességgel);
 - c) a *Profit* típusa legyen valuta ("*Currency*")!

3. Készítsünk SAS adattáblát az *EMPLOYEE_INFO.XLS* fájl *Addresses* táblázatából! A *Postal_Code* típusát módosítsuk karakteresre ("*String*")! Az összes változó esetén a címkézésében *_* helyett space szerepeljen! A *Task*, illetve az output tábla neve egyaránt *Import Employee Addresses* legyen!

4. Készítsünk SAS adattáblát az *EMPLOYEE_PAYROLL.CSV* fájlból! A *Marital_Status* ne szerepeljen a változók között! Az *Employee_ID* típusa legyen numerikus, a *Salary* változónak pedig pénznem! Változtassuk meg a *Birth_Date* kimeneti formátumát *DDMMYYDw.d*-re 10 szélességgel! A kapott táblázatot nevezzük el *Import Employee Payroll*-nak!

3. Műveletek (Task-ok)

1. Határozzuk meg kategóriánként az eladott termékek számát! Használjuk ehhez a *PRO-DUCTS* adattáblát! (Címe: *Number of Products per Category*, Task neve: *Jobs by Dept Freq Report*, keletkezett adattábla neve: *ProductCounts*)

2. Vizsgáljuk meg az *EMPLOYEE_ORGANIZATION* adattábla részlegek (*Department*) változóját megoszlásai gyakoriság és százalékos arány szerint! Az elemzés tartalmazzon egy vízszintes oszlopdiagrammot is! (Címe: *Employee Counts by Department*, Task neve: *Dept Freq Report*) Miután elkészültünk módosítsuk úgy a *Task*-ot, hogy csak a gyakoriságok és a százalékos megoszlások szerepeljenek a riportba!

3. Készítsük el az *EMPLOYEE_ORGANIZATION* adathalmaz alapján a részlegenkénti (*Department*) beosztások (*Job_Title*) gyakoriságát és százalékos arányát! Figyeljünk arra, hogy a gyakoriságok csökkenő sorrendben szerepeljenek a riportba! (Címe: *Employees Job Title*, Task neve: *Jobs by Dept Freq Report*)

4. A *CUSTOMER* adattáblát használva készítsünk országokénti vásárlói listákat, amelyek tartalmazzák a vásárlói azonosítókat (*Customer_ID*), neveket (*Customer_Name*) és a korosztályokat (*Customer_Age_Group*)! (Címe: *Orion Star Customer List*) Ezt követően pedig hajtsuk végre az alábbi módosításokat:

- a) a sorszám helyett a vásárlói azonosító (*Customer_ID*) legyen az első oszlop;
 - b) a *Customer_Name* címkézését módosítsuk Name-re;
 - c) a *Customer_Age_Group* címkézését módosítsuk Age Group-ra;
 - d) a Task neve legyen Customer List by Country!
5. Egy speciális promócióhoz szükségünk van a *CUSTOMER* adattábla alapján egy olyan vásárlói listára, amelyben csak az 1970-ben vagy azután születettek szerepelnek és ráadásul még az Orion Club legaktívabb tagjai is (high-activity members)! A listában csak a *Customer_ID* (sor azonosító), *Customer_Name* és a *Customer_Type* változók szerepeljenek. (Címe: High Activity Customers Born 1970 or Later, Task neve: High Act/1970+) Készítsünk egy magyarázó jegyzetet a Task-hoz!
6. A menedzser szeretne látni az *EMPLOYEE_ADDRESSES* adathalmazból egy listát, amely tartalmazza az *Employee_ID* (sor azonosító), *Employee_Name*, *Street_Number*, *Street_Name*, *City*, *Postal_Code* változókat, melyeknek a nevei *_* nélkül szerepeljenek a listába. Ezt követően hajtsunk végre egy módosítást oly módon, hogy csak a San Diego-i alkalmazottak adatai szerepeljenek a listába a *City* változó feltüntetésével nélkül! Az új lista címe legyen Employee List for San Diego, a Task neve pedig San Diego List!
7. A menedzser szeretne látni az *EMPLOYEE_ADDRESSES* adathalmazból egy listát, amely tartalmazza az *Employee_ID* (sor azonosító), *Employee_Name*, *Street_Number*, *Street_Name*, *Postal_Code* változókat, melyeknek a nevei *_* nélkül szerepeljenek a listába. Ezt követően hajtsunk végre egy módosítást oly módon, hogy csak az ausztráliai alkalmazottak adatai szerepeljenek a listába a *City* változó szerint csoportosítva (sor azonosító)! Az új lista címe legyen Employee List for Australia, a Task neve pedig AU List!

4. Lekérdezések (Query Builder használata)

1. Az *ORDERS* adattáblát felhasználva készítsünk egy új adattáblát, amelyben a 2010 utáni internetes eladások (*Order_Type=3*) szerepelnek rendelési idő (*Order_Date*) szerint csökkenő sorrendben! Az új tábla tartalmazza az összes változót az *Employee_ID* kivételével! A Task, illetve az output tábla neve egyaránt InternetOrders2010+ legyen!
2. Az *EMPLOYEE_ADDRESSES* adattáblát felhasználva készítsünk egy új adattáblát, amelyben a San Diego-i alkalmazottak szerepelnek az irányítószámuk (*Postal_Code*)

szerint növekvő sorrendben! Az új tábla tartalmazza a következő változókat: Employee_ID, Employee_Name, Street_Number, Street_Name, Postal_Code! A Task, illetve az output tábla neve egyaránt SanDiegoEmployees legyen!

3. Az *EMPLOYEE_ORGANIZATION* adattáblát felhasználva készítsünk egy új adattáblát, amelyben az értékesítők (Sales) szerepelnek részlegek (Department) szerint rendezve! Az új tábla tartalmazza az kiindulási tábla összes változóját! A Task, illetve az output tábla neve egyaránt Sales_Emps!
4. Az *EMPLOYEE_ADDRESSES* adattáblát felhasználva készítsünk egy új adattáblát, amelyben azon San Diego-i alkalmazottak szerepelnek, akiknek az irányítószámuk 920-al kezdődik (segítség:substr függvényt kell használni)! Az új tábla tartalmazza a következő változókat: Employee_ID, Employee_Name, Street_Number, Street_Name, Postal_Code! A Task, illetve az output tábla neve egyaránt Postal_Code920_Emps! Végezetül pedig rendezzük irányítószám (Postal_Code) szerint növekvő sorrendbe az adatokat!
5. Az *EMPLOYEE_MASTER* adattáblából készítsünk egy olyan résztáblát, amelyben a 100.000\$-nál kevesebbet kereső ausztráliai alkalmazottak szerepelnek a nevek (Employee_Name) szerint rendezve! Az új tábla tartalmazza a következő változókat: Employee_ID, Employee_Name, Employee_Hire_Date, Salary, City, Department, Job_Title! A Task, illetve az output tábla neve egyaránt AU_Emps legyen! A feladatot a Data Explorer segítségével hajtsuk végre! Válaszoljunk az alábbi kérdésekre
 - a) Az alkalmazottak hány százaléka dolgozik az értékesítési részlegen (Sales Department)?
 - b) Milyen városokban folynak a munkák?
 - c) Mennyi az átlag fizetés?
 - d) Mi a legrégebbi szerződési dátum?

Adjuk hozzá a Data Explorer-t a munka folyamathoz!

6. Az *ORDERS* adattáblából készítsünk egy olyan új adattáblát, amely tartalmazza a szállítási gyorsaságot (hány nap telik el a megrendelés és a kiszállítás között), valamint a teljes kiszámlázott összeget! Az új tábla változói Days_to_Deliver (címkézése: Days to Deliver) és az Invoice_Amt (címkézése: Invoice Amount) mellett az Order_ID, Order_Date, Delivery_Date, Product_ID, Total_Retail_Price, Shipping és a Profit legyenek! Az Invoice_Amt új változónk pénznembe (dollár) szerepeljen 10 szélességgel, két tizedes helyiértéket megengedve! A Task neve a Shipping Detail Query, a tábla neve pedig a Shipping legyen! Végezetül pedig rendezzük a kiszállítási idő (Days_to_Deliver) szerint csökkenő sorrendbe az adatokat!

7. Az *EMPLOYEE_PAYROLL* adattáblából készítsünk egy olyan új adattáblát, amely tartalmazza az alkalmazottak 2%-os fiztésemelését! Az új tábla változói a *New_Salary* mellett *Employee_ID*, *Employee_Gender*, *Salary*, *Birth_Date*, *Employee_Hire_Date* legyenek! A *Salary* változó nevét módosítsuk *Old_Salary* elnevezésre! A *Task* neve a *New Salary Query*, a tábla neve pedig a *New_Salary* legyen! Csak a jelenleg is dolgozó alkalmazottak szerepeljenek a táblában! Az új változónk pénznembe (dollár) szerepeljen 12 szélességgel, két tizedes helyiértéket megengedve!
8. Az *EMPLOYEE_DONATIONS* adattáblából készítsünk egy olyan új adattáblát, amely tartalmazza az alkalmazottak teljes hozzájárulását! Az új tábla változói a *Total_Donations* mellett *Employee_ID*, *Recipients* és *Paid_By* legyenek! A *Task* neve a *Donations Query*, a tábla neve pedig a *Total_Donations* legyen! Az új változónk pénznembe (dollár) szerepeljen két tizedes helyiértéket megengedve!
9. Az *EMPLOYEE_PAYROLL* adattáblából készítsünk egy olyan új adattáblát, amely tartalmazza a cégnél eltöltött napok számát a jelenlegi alkalmazottak esetén (segítség: *yrdif* függvényt kell használni)! Az új tábla változói a *Years_Employed* mellett az *Employee_ID* és *Employee_Hire_Date* legyenek! A *Task* neve a *Years Employed Query*, a tábla neve pedig a *Years_Employed* legyen! Az új változónk típusa numerikus legyen (*w.d*) 4 szélességgel, egy tizedes helyiértéket megengedve!
10. Az *ORDERS* adattáblát felhasználva készítsünk egy olyan táblát, amelyben az összegben 500 \$-nál magasabb nyereségű termék szerepelnek! Az új táblába a teljes profit mellett csak a *Product_ID* változó szerepeljen! Az adatokat a teljes profit alapján rendezzük csökkenő sorrendbe! A *Task* neve a *Top Products Query*, a tábla neve pedig *TopProducts* legyen!
11. Az *EMPLOYEE_MASTER* adattáblát felhasználva készítsünk egy új adattáblát, amelyben az átlagfizetések láthatóak városonként! Az új tábla változói csak a *City* és a *Salary* legyenek! Az adatokat az átlagfizetés alapján rendezzük csökkenő sorrendbe! A *Task* neve a *Average Salary per City Query*, a tábla neve pedig *AvgSal_City* legyen!
12. Az *EMPLOYEE_MASTER* adattáblát felhasználva készítsünk egy új adattáblát, amelyben részlegenként a dolgozók létszáma, átlagfizetése és teljes fizetése látható! A táblába csak azon részlegek szerepeljenek, melyekben 10-nél többen dolgoznak! Az adatok a részlegek nagysága szerint csökkenő sorrendben szereplejenek! A *Task* neve a *Salary Summary by Dept Query*, a tábla neve pedig *Salary_Summary* legyen!
13. Az *EMPLOYEE_MASTER* adattáblát felhasználva készítsünk egy új adattáblát, amelyben a tíz legjobban fizető nem vezetői ("Chief" kiszűrése) munkakör szerepel! Minden

munkakörhöz számoljuk ki az átlagfizetést és állítsuk a formátumát pénznemre (dollar)! Az adatokat az átlagfizetés alapján rendezzük csökkenő sorrendbe! A Task neve legyen Top10 Paid Job Titles!

14. A menedzser szeretné részletesebben megvizsgálni azon termékeket, amelyek teljes profitja 500\$-nál nagyobb. Az ehhez szükséges adatok három különböző táblázatban találhatóak. Feladatunk így a *TOPPRODUCTS*, a *PRODUCTS* és a *COUNTRY_LOOKUP* adattáblák egyesítése a Product_ID és a Supplier_Country–Country_Key kulcsok segítségével. Az egyesített új tábla változói a Product_ID, a SUM_of_Profit, Product_Category, Product_Name, Supplier_Name és a Country_Name legyenek! A Task neve a Top Products Info Query, a tábla neve pedig Top_Products_Info legyen!
15. Készítsünk az *EMPLOYEE_PAYROLL* és az *EMPLOYEE_ADDRESSES* adattáblákból egy új táblázatot az Employee_ID változó szerint egyesítve! Az egyesített új tábla változói az Employee_ID, Employee_Name, Employee_Gender, Birth_Date, Salary, Street_Name, Street_Number, City, State és a Country legyenek! A Salary változó formátumát állítsuk pénznemre (dollar) 12 szélességgel, két tizedes helyiértéket megengedve, a Birth_Date változót pedig dátumra (DATEw.) 9 szélességgel beállítva! Készítsünk egy Bonus elnevezésű új változót, amely a fizetések 1,5%-ként keletkezik és legyen a formátuma pénznem (dollar) 10 szélességgel, két tizedes helyiértéket megengedve! Csak a jelenleg is dolgozó alkalmazottak kerüljenek be az új táblába! A Task neve a Payroll Location Join Query, a tábla neve pedig Employee_Payroll_Location legyen! Módosítsuk az elkészült Query-t oly módon, hogy csatoljuk hozzá a *COUNTRY_LOOKUP* adattáblát a Country–County_Key változók segítségével, valamint a Country változót cseréljük le a teljes neveket tartalmazó Country_Name változóra!
16. Kapcsoljuk össze az *EMPLOYEE_ADDRESSES* és az *EMPLOYEE_DONATIONS* adattáblákat! Figyeljünk arra, hogy az *EMPLOYEE_ADDRESSES* táblából minden sor és minden oszlop bekerüljön az új táblába! Az új táblába csak azok az alkalmazottak szerepeljenek, akik nem adakoztak (azaz, ahol az *EMPLOYEE_DONATIONS* tábla Employee_ID változója üres)! A Task neve az Employee Without Donations Query, a tábla neve pedig No_Donations legyen!

5. Grafikonok szerkesztése

1. A *CUSTOMER* adattábla alapján készítsünk egy vízszintes háromdimenziós oszlopdiagramot a vásárlók országonkénti átlagos életkorára vonatkozóan! Ezt követően módosítsuk a diagramunkat úgy, hogy az átlagéletkor formátuma numerikus (w.d) legyen 4 szélességgel, két tizedes helyiértéket megengedve, az oszlopok alakja kör legyen és szagatott referencia vonalak (20-40-60 évnél meghúzva) segítsék a könnyebb leolvasást!

Az x tengely címét is módosítsuk egyúttal Average Customer Age feliratra, az y tengelyét pedig Country-ra! Az ábra címe Average Customer Age by Country legyen, a Task neve pedig Avg Age/Country! A SAS riport mellett HTML formátumban is készítsük el a grafikont!

2. Az *EMPLOYEE_DETAIL* adattábla alapján készítsünk egy háromdimenziós oszlopdiagramot a városenkénti teljes fizetésekre vonatkozóan (y tengely címe: Total Salary)! Minden oszlop legyen különböző színű és a teljes fizetések leolvashatóak legyenek az oszlopok tetejéről! Ezt követően módosítsuk a diagramunkat úgy, hogy a fizetés formátuma egészre kerekített érték legyen, az oszlopok alakja prizma és világos szürke, szagatott referencia vonalak segítsék a könnyebb leolvasást! Az ábra címe Total Salary for Each City legyen, a Task neve pedig Total Salary/City! A SAS riport mellett HTML formátumban is készítsük el a grafikont!

3. Az *EMPLOYEE_DETAIL* adattábla alapján készítsünk egy háromdimenziós kördiagramot a részlegenkénti teljes fizetésekre vonatkozóan (a körcikkek belsejében lássuk a százalékos megoszlásokat, kívül pedig a fizetéseket és a részlegek neveit)! Ezt követően módosítsuk a diagramunkat úgy, hogy a fizetés formátuma egészre kerekített érték legyen! Az ábra címe Total Salary by Department legyen, a Task neve pedig Total Salary/Dept! A SAS riport mellett HTML formátumban is készítsük el a grafikont!

4. Készítsük el az alábbi öt indiai állam adatait tartalmazó táblázat alapján az adatrögzítést Excelben! Importáljuk be EG-be az elkészített táblázatot és nevezzük el Population Data of India-nak!

States	Population (million)	Distance Travelled by Train (km)	Gross Domestic Product of State (GSDP)
Punjab	28,16	2156	1565
Tamil Nadu	68,11	3943	4165
Uttar Pradesh	206,31	8800	4200
Uttarakhand	10,17	345	609
West Bengal	90,8	4000	3336

- a) Szemléltessük államonként a GSDP és Rail_km változókat Scatter Plot segítségével (x tengely címe: States; y tengely címe: Rail Distance (km); y tengely (jobb oldal) címe: Gross Domestic Product; ábra címe: Scatter Plot of Distance Travelled by Train and GSDP of Five States in 2012)!
- b) Szemléltessük államonként a GSDP és Rail_km változókat oszlop-vonal (Bar-Line Chart) segítségével! Használjuk az a) részben megadott címeket!
- c) Szemléltessük államonként a GSDP és Rail_km változókat csempe ábra (Tile Chart) segítségével! Használjuk az a) részben megadott címeket!

d) Szemléltessük államonként mindhárom változót buborék ábra (Bubble plot) segítségével (ábra címe: Population, Distance travelled by rail and GSDP of five States)! A tengelyeket értelemszerűen nevezzük el!

5. Készítsük el az alábbi Magyarország megyéinek népsűrűségi adatait tartalmazó táblázat alapján az adatrögzítést Excelben! Importáljuk be EG-be az elkészített táblázatot és nevezzük el Hu_Nepsuruseg-nek! Szemléltessük az adatokat egy háromdimenziós térkép ábrán (ábra címe: Magyarország megyéinek népsűrűsége)!

ID	Megyék	Nepsuruseg (fő/km ²)
2	Baranya	89,5
4	Borsod-Abaúj-Zemplén	97,9
5	Budapest	3305,23
1	Bács-Kiskun	63,1
3	Békés	66,8
6	Csongrád	99,5
8	Fejér	98,3
9	Győr-Moson-Sopron	108,7
10	Hajdú-Bihar	96,5
11	Heves	95,6
20	Jász-Nagykun-Szolnok	71,5
12	Komárom-Esztergom	139
14	Nógrád	82,6
16	Pest	190
17	Somogy	53,8
18	Szabolcs-Szatmár-Bereg	93
21	Tolna	64,3
22	Vas	78
23	Veszprém	78,4
24	Zala	77

6. Összegző táblák, jelentések készítése

1. Az *ORION_PROFIT* adattábla alapján készítsünk egy profit összegző riportot a termékek fajtája (Product_Line) és kategóriája (Product_Category) szerint! A riportban csak az átlagok, mediánok és az összegek szerepeljenek két tizedes helyiértéket megengedve! Szemléltessük a kapott eredményeket dobozábrán (Box and whisker)! Az elkészített statisztikákat mentjük el táblázatba is profit_summary névvel és exportáljuk ki Excel-be! A riport címe Summary of Profit by Product Line/Category, a Task neve pedig Profit by Product Summary legyen!

2. Az *EMPLOYEE_MASTER* adattábla alapján részlegenként készítsünk elemzést az alkalmazottak fizetéséről! A riportban csak az átlagok, mediánok, legkisebb és leg-

nagyobb értékek szerepeljenek kerekítve (egész dollárban)! Az elkészített statisztikákat mentsük el táblázatba is salary_stats névvel és exportáljuk ki Excel-be (SalaryStats.xlsx)! A riport címe Summary Statistics for Salary by Department, a Task neve pedig Salary by Dept Summary legyen!

3. Az *EMPLOYEE_MASTER* adattábla alapján részlegenként és államonként készítsünk elemzést az alkalmazottak fizetéséről! A riportban csak az átlagok, mediánok, legkisebb és legnagyobb értékek szerepeljenek kerekítve két tizedes helyiértékig! A riport címe Summary Statistics for Salary by Department and State, a Task neve pedig Salary by Dept/State Summary legyen! Ezt követően módosítsuk úgy a Taskot, hogy a két változó (részlegek és államok) szerint minden lehetséges módon készítsük el a fizetések elemzését!

4. Az Orion marketing csoport annak érdekében, hogy jobban megértse a vásárlók szokásait szeretne látni egy az *ORION_PROFIT* adattábla alapján készített összegző táblát a teljes profitról a vásárlók korosztálya és termékek kategóriája alapján. A profit változó pénznembe (dollár) szerepeljen 8 szélességgel! Ezt követően hajtsuk végre az alábbi módosításokat:
 - a) Az összegző táblázat fejlécébe az elemző változó (Profit) és a statisztikai függvény (Sum) ne szerepeljen!
 - b) A termékek fajtáin (Product_Line) belül kategóriák (Product_Category) szerint is összegezzünk!
 - c) A Product_Category és Product_Line kiírások ne szerepeljenek a táblázatba!
 - d) Készítsünk összegzést a termékek minden fajtája esetén is!
 - e) Állítsuk be a Profit by Customer Age Group és a Product Line Subtotal elnevezéseket a fejlécekben!
 - f) A Product Line Subtotal felirat legyen dőlt betűs (Bold Italic) és világos sárga (light yellow) háttérű!
 - g) Hiányzó adatok helyére ** kerüljön!
 - i) A riport címe és Task neve egyaránt Profit by Category and Line legyen!

5. Készítsünk egy összegző táblázatot az *EMPLOYEE_MASTER* tábla adataiból, amelyben a részlegenkénti legkisebb és legnagyobb fizetések szerepelnek országonként lebontva! A fizetések pénznembe (dollár) szerepeljenek 10 szélességgel! A táblázat oszlopaiiban az országok, soraiban pedig a részlegek (fejlécben ne legyen feltüntetve) szerepeljenek! Az összegző táblázat fejlécébe az elemző változó (Salary) ne szerepeljen! Ezt követően hajtsunk végre az alábbi módosításokat:

- a) A Minimum és Maximum feliratokat cseréljük ki Low, illetve High elnevezésekre!
- b) Az oszlopok Country elnevezésű fejlécét írjuk át Salary Ranges by Country-ra!
- c) Hiányzó adatok helyére "-" kerüljön!
- d) A riport címe és Task neve egyaránt Salary Ranges by Country legyen!

6. A menedzser szeretne látni az *EMPLOYEE_MASTER* adathalmazból egy összegző táblázatot, amelyben az összes Ausztrál városra felbontva részlegenként szerepelnek a fizetések gyakoriságai és %-os gyakoriságai. A táblázat oszlopaiban a városok (fejlécben ne legyen feltüntetve), soraiban pedig a részlegek (fejlécben ne legyen feltüntetve) szerepeljenek! A összegző tábla címe legyen Summary of Salaries for Australia by Department, a Task neve pedig AU Salary Report! Ezt követően hajtsunk végre az alábbi módosításokat:

- a) Az összesítő oszlopból (Total) töröljük a ColPctN-t!
- b) Legyen mind az összesítő sor (Total) és mind az összesítő oszlop (Total) háttere világos narancssárga (light orange)!

7. Prompt-ok használata

1. Az *ORION_PROFIT* adattáblát felhasználva készítsünk egy kategóriánkénti (Product_Category) profit eloszlást szemléltető háromdimenziós kördiagram promptot (típusa: variable, neve: ProfitCategories)! A körcikkekben legyen feltüntetve a százalékos megoszlás! A grafikon címe Profit by &ProfitCategories legyen! A promptot teszteljük a kategóriákat (Product_Category) követően az országokra (Customer_Country) is!
2. Az *EMPLOYEE_MASTER* adattáblát felhasználva készítsünk az átlagfizetést szemléltető háromdimenziós vízszintes oszlopdiaagram promptot az alkalmazottak neme (Employee_Gender), városa (City), országa (Country) és részlege (Department) szerint (prompt típusa: variable , prompt neve: SelectVar, felugró szöveg: Which column do you want to analyze?)! Minden oszlop legyen különböző színű és az átlag fizetések leolvashatóak legyenek az oszlopok végéről! Az x tengely címe Average Annual Salary, a grafikon címe pedig Average Annual Salary by &SelectVar legyen! A promptot teszteljük le az összes megadott változóra!
3. Az *ORION_PROFIT* adattáblát felhasználva készítsünk a szállítók rendeléseinek (Supplier_Name) szűrésére alkalmas promptot (prompt típusa: text , prompt neve: ChooseSupplier, felugró szöveg: Select a Supplier:)! Válasszuk ki a Product_ID, Product_Name, Order_Date, Quantity és Profit változókat! Az elkészített lista címében mindig a megfelelő szállító szerepeljen (Orders for &ChooseSupplier)! (Task neve:

Choose a Supplier Query, elkészített tábla neve: SupplierOrders)

4. Az *EMPLOYEE_MASTER* adattáblát felhasználva készítsünk egy városok (City) szűrésére alkalmas promptot (prompt típusa: text, prompt neve: SelectCity, felugró szöveg: Select a City:)! Válasszuk ki az összes változót! Alapértelmezett városnak a Miami-Dade legyen beállítva! A promptot teszteljük Sydney városára is! (Task neve: Select City Query, elkészített tábla neve: select_city)
5. Az *EMPLOYEE_MASTER* adattáblát felhasználva készítsünk egy szerződési dátum (Employee_Hire_Date) szűrésére alkalmas promptot (prompt típusa: date range, prompt neve: DateRange, felugró szöveg: Select a beginning and end date for the employee hire date range:)! Válasszuk ki az összes változót! Az eredmények legyenek az Employee_Hire_Date alapján növekvő sorrendbe rendezve! A promptot teszteljük 1976. január 1-től 1989. december 31-ig! (Task neve: Employee Hire Date, elkészített tábla neve: employee_hire_date)
6. Az *EMPLOYEE_MASTER* adattáblát felhasználva készítsünk egy városok (City) szűrésére alkalmas többszörös értéket bekérő promptot (prompt típusa: text, prompt neve: SelectCities, felugró szöveg: Select a Cities to Include:)! Válasszuk ki az összes változót! A promptot teszteljük Melbourne és Sydney városokra! (Task neve: Multi_Cities Query, elkészített tábla neve: Multi_Cities)

8. Projekt eredmények testreszabása és rendezése

1. Az Orion Star elemzőjének szüksége van egy összefoglaló riportra a Profit Analysis projekt eredményeiből (Profit by Cust Age Group, Profit by Product Category Summary, Profit by Cust Cntry/Type). Ennek elkészítéséhez használjuk a *e107d01.egp* projektet! A riportot ruházzuk fel az Orion-Banner.jpg szalagcímmel (kék színű, dupla vonallal elválasztva az eredményektől)! Az oldal elrendezése legyen Landscape! A riport task neve Profit Report legyen! Az összegző riportot készítsük el HTML formátumban is!
2. Az Orion Star elemzőjének szüksége van egy összefoglaló riportra az Employee Analysis projekt eredményeiből (One-Way Frequencies, Summary Tables, Bar Chart). Ennek elkészítéséhez használjuk a *e107e01.egp* projektet! A riportot ruházzuk fel az Orion-Banner.jpg szalagcímmel! Az oldal elrendezése legyen Landscape! A riport task neve Employee Report legyen! Az összegző riportot készítsük el PDF formátumban is!
3. Rendezzük át az *e107d02.egp* projektet úgy, hogy minél áttekinthetőbb legyen!

4. Rendezzük át az *e107e02.egp* projektet úgy, hogy minél áttekinthetőbb legyen! Ezt követően adjunk hozzá egy új munkafolyamatot a projekthez Graphic Reports névvel! A Graphic Reports munkafolyamathoz rendeljük hozzá az oszlopdiagram (Bar Chart) és kördiagram (Pie Chart) Taskokat és eredményeket, ezt követően pedig frissítsük az eredményeket!

Irodalom

- [1] Jánosa András, *Üzleti intelligencia alkalmazások*. ComputerBooks, Budapest, 2010.
- [2] Olivia Parr-Rud, *Business Analytics Using SAS Enterprise Guide and SAS Enterprise Miner: A Beginner's Guide*. SAS Institute, 2014.
- [3] SAS OnlineDoc 9.4 for the Web—<https://support.sas.com/documentation/onlinedoc/guide/>