Univerzitet u Nišu,

Elektronski fakultet Niš

Logo

Description automatically generated

*Seminarski rad*

Sistemi za upravljanje bazama podataka

**BACKUP RESTORE KOD MICROSOFT SQL SERVER-A**

Mentor: Student:

Aleksandar Stanimirović Ivan Milojković 1687

Niš, sepembar 2024. godine

**Sadržaj**

[**1.** **Uvod**…………………………………………………………………3](#_Toc135491158)

[**2.** **Tipovi rezervnih kopija**……………………………………………4](#_Toc135491159)

**[3.](#_Toc135491165)****[Kompresija i enkripcija rezervnih kopija](#_Toc135491165)** […………………………….. .15](#_Toc135491165)

**[4. Restore baze podataka](#_Toc135491165)** [16](#_Toc135491165)  
**5. Planovi za oporavak od katastrofe**……………………………………….22

[**6.** **Zaključak** 23](#_Toc135491166)

[**7.** **Literatura** 25](#_Toc135491167)

# **Uvod**

Koliko podaci znace mozemo da vidimo u svakodnevnom zivotu. Od skoro je izasao clanak u kome se navodi da su podaci najvredniji resurs na svetu, čak vredniji i od nafte i njenih sirovina. Oni su ključni za poslovanje jer omogućavaju kompanijama bolje razumevanje svojih klijenata i tržišta, što im pomaže u daljem razvoju i unapređenju proizvoda i usluga. Bez njih, kompanije ne bi mogle optimizovati svoje procese, povećati produktivnost niti smanjiti troškove. Zbog svega toga je zaštita podataka izuzetno važna, jer njihov gubitak može dovesti do ozbiljnih posledica.

Backup predstavlja proces izrade rezervne kopije podataka kako bi se podaci zaštitili od gubitka usled oštećenja, kvarova hardvera ili softvera, ili drugih nepredviđenih događaja. Backup proces obično podrazumeva kreiranje redovnih kopija podataka u planiranim vremenskim okvirima , svakodnevno ili nedeljno, i čuvanje tih kopija na bezbednim mestima.

Sistem za upravljanje bazama podataka (DBMS) omogućava korisnicima da kreiraju, uređuju, skladište i upravljaju podacima u bazama. DBMS je sastavljen od niza programa koji korisnicima omogućavaju pouzdano i efikasno rukovanje velikim količinama podataka. On je zadužen za očuvanje sigurnosti podataka pružajući sigurnosne mehanizme za zaštitu i kontrolu pristupa. Takođe, omogućava korisnicima pravljenje rezervnih kopija podataka, a u slučaju gubitka podataka, omogućava im da brzo obnove podatke iz tih kopija i vrate bazu u prethodno stabilno stanje

Kreiranje rezervne kopije podataka podrazumeva pravljenje kopije podataka i njeno skladištenje na sigurnom mestu, kao što su eksterni hard diskovi ili cloud. Posedovanje važećih rezervnih kopija baze podataka ključno je za oporavak u slučaju problema.

Osim što se koristi za vraćanje podataka u slučaju gubitka, rezervna kopija podataka služi i za kopiranje baze podataka sa jednog servera na drugi. Ovo je posebno korisno kada je potrebno prebaciti bazu podataka na novi server ili kreirati kopiju za potrebe testiranja baze.

Vraćanje ili restore baze podataka uključuje oporavak baze podataka iz rezervne kopije i vraćanje na njeno prethodno validno stanje. Kreiranje rezervne kopije i vraćanje podataka se treba prilagoditi odredjenom okruzenju i raspoloživim resursima. S toga, jako je bitno da se prati dobro osmišljena strategija prilikom kreiranja rezervnih kopija i vraćanje (restore) podataka.

Takva strategija bi trebala da se sastoji iz dela za kreiranje rezervne kopije i dela za vraćanje podataka. Deo strategije za backup definiše tip i učestalost pravljenja rezervnih kopija, prirodu i brzinu hardvera koji je zadužen za njih, način na koji se testiraju kreirane kopije, kao i gde i kako treba čuvati rezervne kopije. Deo strategije za restore definiše ko je odgovoran za obavljanje vraćanja, kako se vraćanje treba izvršiti kako bi se postigli ciljevi vezani za dostupnost baze podataka i minimiziranje gubitka podataka, kao i način na koji se restore proces testira. Važno je da obe strane strategije budu pažljivo osmišljene i testirane kako bi se obezbedila pouzdana sigurnost podataka i efikasan oporavak.

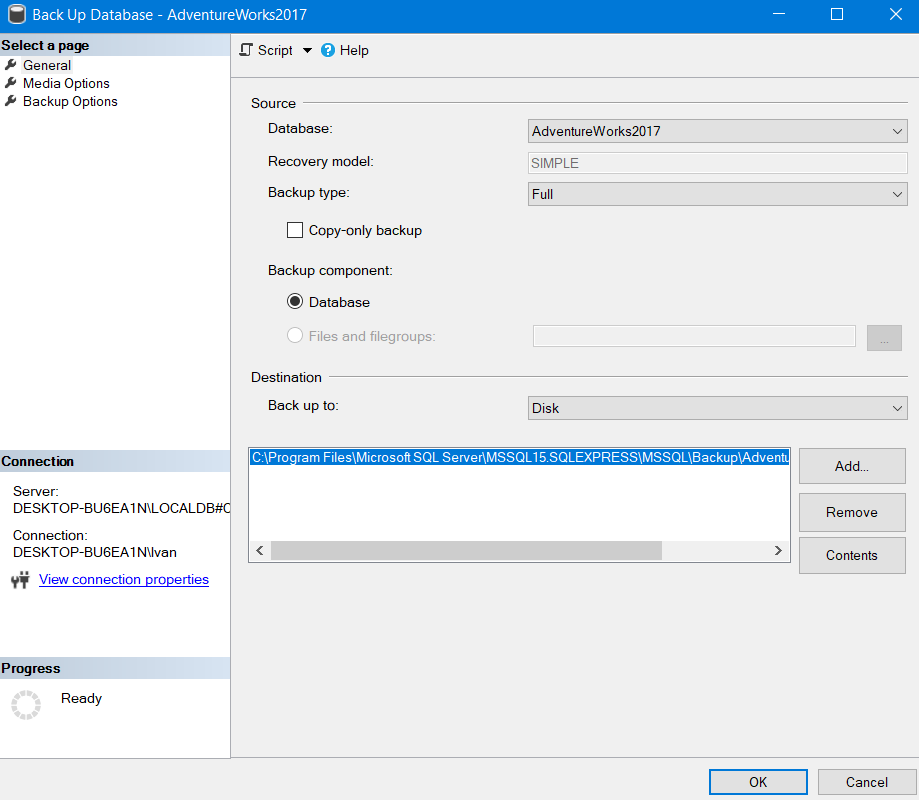
Da bi se osmislila efikasna strategija pravljenja rezervne kopije i vraćanja, potrebno je pažljivo obratiti paznju na planiranje, implementaciju i samo testiranje. Definisanje odgovarajuće strategije je složen proces koji zahteva pažljivo razmatranje različitih faktora. To uključuje zahteve za sigurnost i očuvanje podataka, učestalost pristupa, kao i obim baze podataka. Takođe je važno uzeti u obzir dostupne resurse poput prostora za skladištenje, tehničku infrastrukturu, ljudske resurse i budžet.

# **Tipovi rezervnih kopija**

U nastavku seminarskog rada, bice obradjene nekoliko vrsti rezervnih kopija:

* Full backup,
* Differential backup,
* Transaction log backup,
* Tail-log backup,
* Copy-only backup,
* File backup,
* Partial backup.

Rezervne kopije je moguće kreirati i pomoću *SQL Server Management Studio*-a (*Database → Tasks → Back Up…*) (*Slika 1.*).

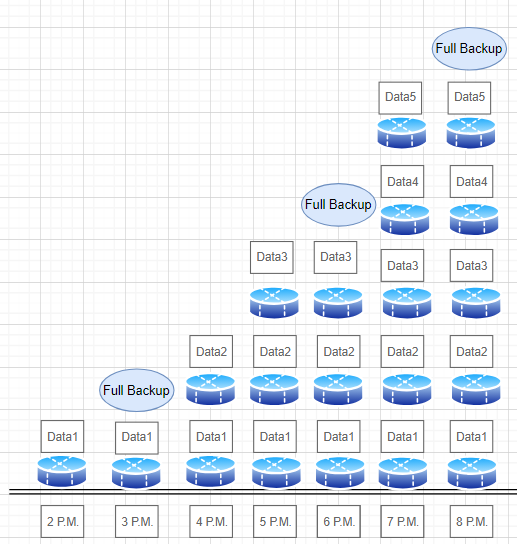


Slika 1. Kreiranje backup-a pomoću SQL Server Management Studio-a

## **Full backup**

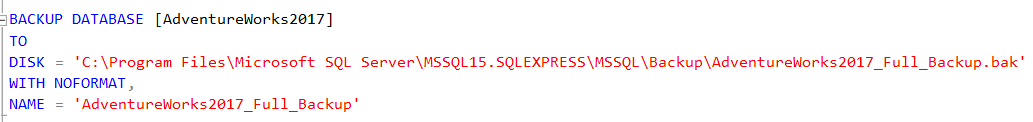
Potpuna rezervna kopija ili full backup kopira celu bazu podataka, tj. sve podatke, kao i: tabele baze podataka, procedure, funkcije, indekse, poglede itd... Ona uključuje i deo transaction log-a, kako bi se baza podataka mogla u potpunosti vratiti na trenutak kada je rezervna kopija napravljena.

Potpuna rezervna kopija predstavlja osnovu za ostale tipove rezervnih kopija i omogućava najprostiji način vraćanja baze podataka, jer se svi njeni sadržaji nalaze u jednoj datoteci. Ona takođe pruža najpotpuniju zaštitu od gubitka podataka. Jedan downside ovog nacina backup-ovanja jeste da kako se vremenom baza povecava, tako se i vreme koje je potrebno za njeno vraćanje povecava, tako da se prlikom backup-ovanja potpune rezervne kopije baze podataka mora uzeti u obzir dobra strategija. Na *slici 2.* možemo videti raspored periodičnog pravljenja potpune rezervne kopije.



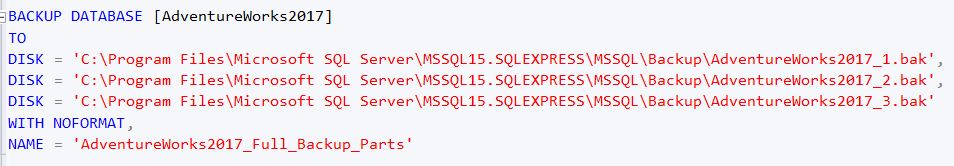
Slika 2. Kreiranje potpune rezervne kopije u nekom sekvencijalnom periodu

Naredba za kreiranje potpune rezervne kopije baze podataka:



Slika 3. Naredba za kreiranje potpune rezervne kopije baze podataka

Kao sto je maločas pomenuto, u situacijama kada baza podataka postane prevelika, rezervnu kopiju možemo podeliti na manje delove. (Slika 3.). Ovo omogućava da proces pravljenja rezervne kopije bude brži i jeftiniji. Podelom rezervne kopije, svaka datoteka sadrži samo deo baze, odnosno njenih podataka, što olakšava skladištenje. Pored toga, smanjuje se rizik od gubitka podataka, jer ako se jedna od datoteka ošteti ili izgubi, preostale datoteke se mogu koristiti za obnavljanje baze podataka.

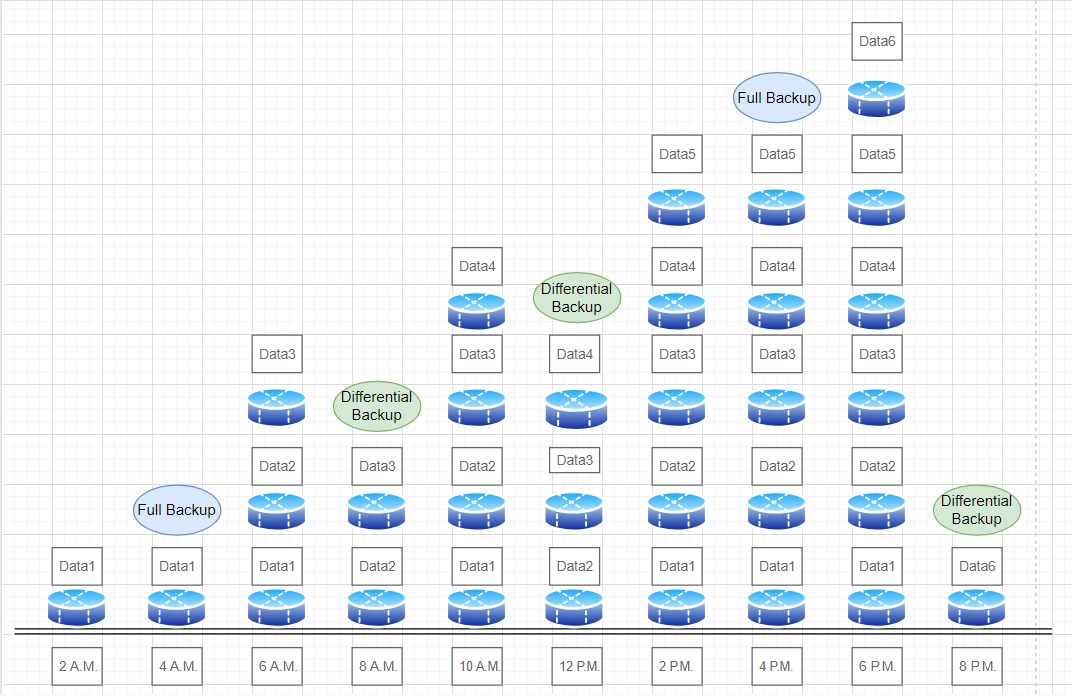


## Slika 4. Naredba za kreiranje potpune rezervne kopije izdeljena u više datoteka

## **Differential backup**

Diferencijalna rezervna kopija baze podataka ima za cilj smanjivanje vremena izrade same rezerve kopije. Proces kreiranja diferencijalne rezervne kopije podataka je kopiranje samo onih stranica koje su dozivele promenu od poslednje potpune promene. Ovakav nacin kreiranja rezervne kopije smanjuje obim podataka koje je potrebno sačuvati, ubrzava proces i štedi prostor za skladištenje, a istovremeno omogućava brži oporavak baze podataka u slučaju gubitka podataka  
  
 Temelj ovakve vrste kreiranja rezervne kopije baze podataka jeste potpuna rezervna kopija, kao i kod nje, kopira se transaction log kako bi se podaci mogli vratiti u potpunosti. Kao što je pomenuto, najveća prednost kod ove vrste backup-a jeste to sto je potrebno manje vremena za samu izradu, zbog toga što se ne kopiraju svi podaci, pa se tako i manje resursi troše.

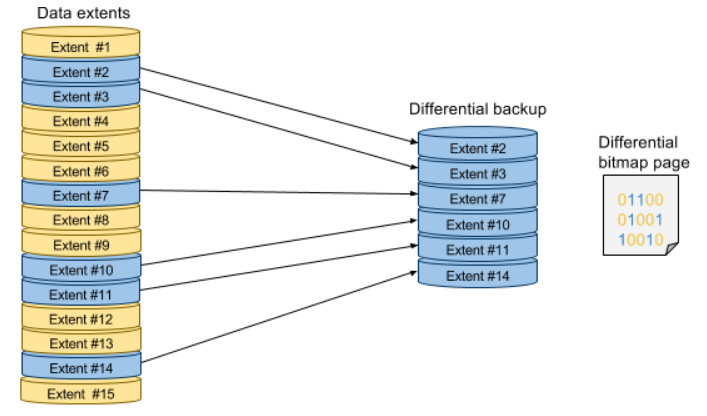
Izrada diferencijalne rezervne kopije može biti mnogo brža u odnosu na izradu potpune rezervne kopije baze podataka. Međutim, prilikom povratka, odnosno restore-a, diferencijalne rezervne kopije, potrebno je prvo vratiti najnoviju potpunu rezervnu kopiju, a zatim primeniti diferencijalnu rezervnu kopiju. Kako veličina diferencijalne rezervne kopije raste prilikom svake sledeće izrade, ona gubi prednost u odnosu na potpunu rezervnu kopiju, a to su brži oporavak podataka i manje iskorišćenih resursa. Samim tim je potrebno periodično izrađivati potpune rezervne kopije, kako bi svaka nova diferencijalna kopija koristila što manje resursa (*Slika 5.*).



Slika 5. Kreiranje diferencijalne rezervne kopije u kombinaciji sa potpunom rezervnom kopijom

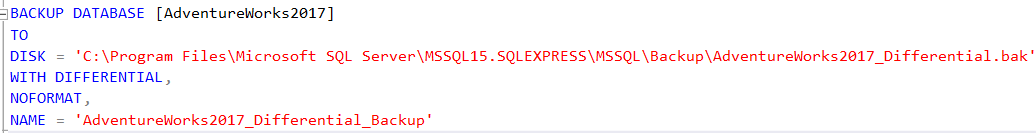
Diferencijalne rezervne kopije baze podataka mogu biti veoma korisne u situacijama kada se određeni delovi baze podataka često menjaju. Jedan primer je ukoliko baza podataka sadrži poprilično veliku tabelu koja se često ažurira, a većina drugih tabela su relativno nepromenljive. Tada, diferencijalna rezervna kopija se može koristiti samo za promene napravljenih u tabeli koja se često ažurira, i na taj način dodatno pomaže gubljenju vremena i resursa.

Diferencijalna rezervna kopija beleži stanje svih *extent*s-a (**Extent** predstavlja osnovnu jedinicu za alokaciju prostora na disku koja se koristi za organizovanje i upravljanje skladištenjem podataka) koje su se promenile od poslednje potpune rezervne kopije. Na *slici 6.* možemo videti princip izrade diferencijalne rezervne kopije baze podataka. Prikazana su 15 **extents-a**, od kojih su 6 označeni kao promenjeni od poslednje potpune rezervne kopije. Kreiranje diferencijalne kopije se oslanja na stranicu polja bitova (eng. bitmap page) koja sadrži jedan bit za svaki *extents*. Ukoliko dodje do promene **extents-a**, bit se postavlja na 1 u stranici polja bitova. Diferencijalna kopija baze podataka se sastoji samo od **extents-a** kojima su bitovi postavljeni na vrednost 1.



Slika 6. Primer kreiranja diferencijalne rezervne kopija

Naredba kreiranja diferencijalne rezervne kopije baze podataka:



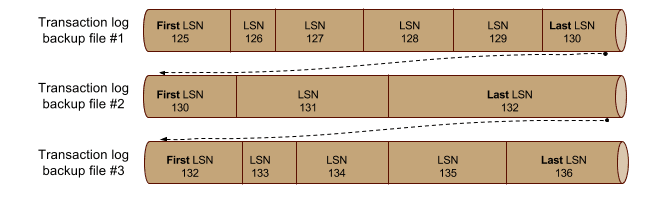
Slika 7. Naredba za kreiranje diferencijalne rezervne kopije

## **Transaction log backup**

Transaction log, tj. evidencija završenih transakcija, predstavlja evidentiranje transakcija, odnosno njihovo skladištenje, i to onih transkacija koje su se izvršile od poslednje potpune rezervne kopije baze podataka (što znači da je potrebno izraditi barem jednu potpunu rezervnu kopiju da bi se mogle evidentirati završene transakcije). Svaka evidencija sadrži detalje koje se odnose na kreiranje i izmenu objekata, kao i svaku izmenu podataka.

Pošto se evidencije izvršenih transakcija pamte sekvencijalno, ukoliko se vrši oporavak podataka, moguće povratiti operacije nad objektima i podacima u istom redosledu kao što su se i izvršile. Zbog toga je moguće vratiti bazu podataka u tačno određeno stanje. Svaka evidencija izvršenih transakcija je ima svoj jedinsteni broj koji se naziva sekvencijalnim brojem evidencije (eng. Log Sequence Number – LSN). Sto je on manji, to je evidencija starija, i suprotno sto je LSN veci to je evidencija bliže onoj najnovijoj.

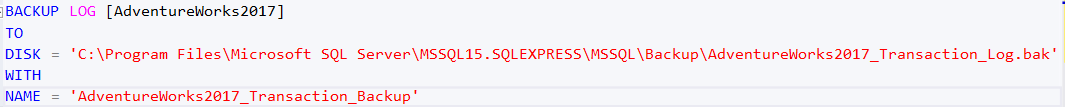
Na *slici 8* . vidimo da su evidencije transakcija povezane lančano. Lanac je neprekidni niz koji sadrži sve evidencije neophodne za oporavak baze podataka do nekog određenog trenutka. Lanac uvek počinje potpunom rezervnom kopijom baze podataka i nastavlja se nizom evidencija izvršenih transakcija.



Slika 8. Lančano skladistenje evidencija izvršenih transakcija

Primarni razlog za izradu evidencija izvršenih transakcija je čuvanje transakcija koje su se dogodile od poslednje potpune rezervne kopije baze podataka, kako bi se pomoću transakcija baza podataka dovela u pređašnje stabilno stanje, ukoliko je došlo do nekih problema, na taj način imamo sigurno zapisano poslednje stabilno stanje.

Naredbu kreiranja transaction log rezervne kopije baze podataka:



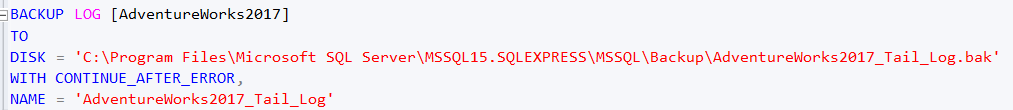
Slika 9. Naredba kreiranja transaction log-a rezervne kopije baze podataka

## **Tail-only backup**

Tail-log backup je predstavlja jednu vrstu transaction log backup-a koja obuhvata sve potvrđene transakcije koje su izvršene od trenutka poslednjeg transaction log backup-a, odnosno poslednji deo transaction log-a koji se kreira kao backup u slučaju da dođe do greške ili kvara na bazi podataka, pre nego što se izvrši proces oporavka.

Kod korišcenja tail-log backup-a, pamte se sve transakcije koje su završene i potvrđene, ali koje nisu bile uključene u prethodnu evidenciju izvršenih transakcija, zbog toga što ne želimo doći u situaciju da transakcije budu izgubljene prilikom oporavka baze podataka. Važno je napomenuti da se ova vrsta rezervne kopije može izvršiti samo kada je baza podataka u režimu potpunog ili bulk-logged oporavka. Ova vrsta rezervne kopije se koristi u situacijama gde je minimalni gubitak podataka od presudnog značaja.

Naredbu kreiranja tail-log rezervne kopije baze podataka:



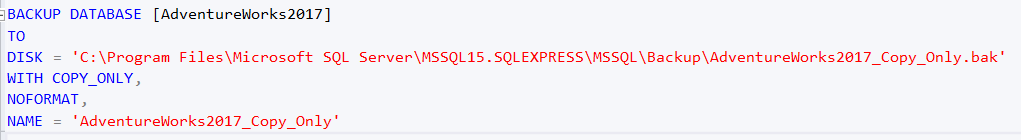
Slika 10. Naredna kreiranja tail-log backup-a

## **Copy-only backup**

Copy-only backup u SQL Serveru predstavlja specifičnu vrstu rezervne kopije koja ne utiče na uobičajeni redosled i strategiju pravljenja drugih rezervnih kopija. Dok standardni backup procesi mogu promeniti način na koji se baza kasnije oporavlja, copy-only backup se koristi kada je potrebno napraviti rezervnu kopiju u posebnim okolnostima, a da pritom ne menja redovni raspored pravljenja i vraćanja ostalih rezervnih kopija. Ova opcija je korisna kada želimo da obavimo backup za određene namene, bez narušavanja standardnog postupka backup-a.

Kada se kreira copy-only backup, baza podataka ostaje neizmenjena na način koji bi uticao na buduće redovne rezervne kopije ili njihov oporavak. Ova vrsta rezervne kopije koristi se u situacijama kada je potrebno napraviti posebnu kopiju, na primer za testiranje, migraciju baze na drugi server ili arhivske svrhe, bez narušavanja postojećih procedura za pravljenje i obnavljanje regularnih rezervnih kopija.

Naredba kreiranja copy-only rezervne kopije baze podataka:

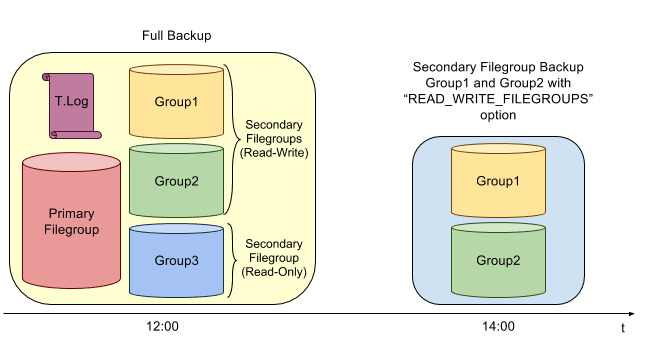


Slika 11. Naredba kreiranja copy-only backup-a

## **File backups**

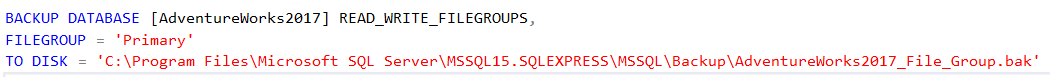
File backup omogućava pravljenje kopije samo određenih datoteka ili setova datoteka koje čine bazu podataka, umesto kompletnog backup-a cele baze. Postoji nekoliko tipova ovakvih backup-ova: potpuni backup datoteka (full file backup), diferencijalni backup datoteka (differential file backup), kao i delimični backup datoteka (partial file backup). Svaka od ovih vrsta rezervnih kopija omogućava fleksibilniji način zaštite podataka u zavisnosti od potreba korisnika.

Potpuni backup datoteka obuhvata sve podatke i objekte unutar određene datoteke ili grupe datoteka. Ova vrsta backup-a omogućava brži proces oporavka, jer se mogu obnoviti samo oštećene datoteke, bez potrebe za vraćanjem cele baze podataka. Takođe, pruža veću fleksibilnost pri kreiranju strategije pravljenja rezervnih kopija, jer omogućava preciznije planiranje i upravljanje backup procesima.



Slika 12. Raspored kreiranja potpune rezervne kopije datoteka nakon potpune kopije

Naredba kreiranja potpune rezervne kopije datoteka:



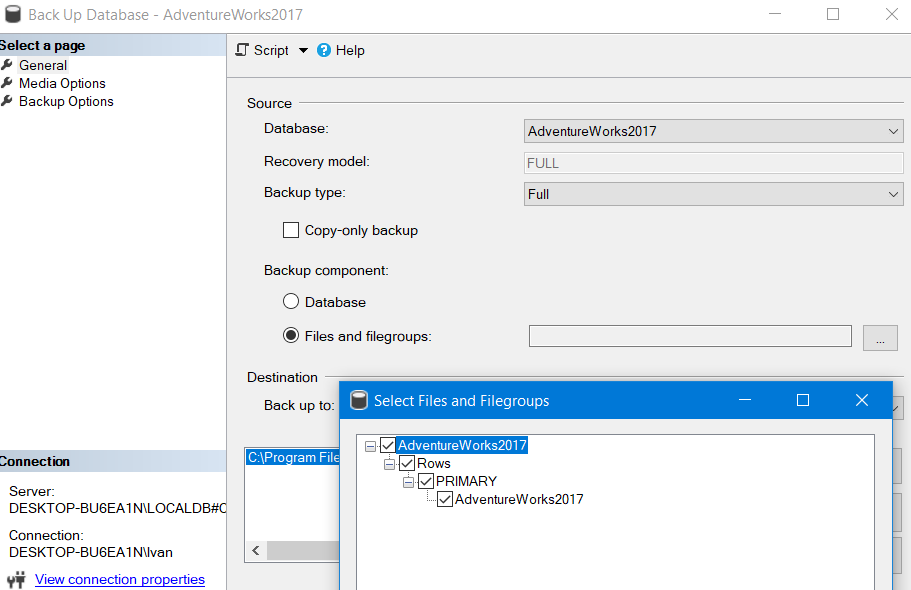
Slika 13. Naredba kreiranja potpune rezervne kopije datoteka

Diferencijalni backup datoteka (differential file backup) pravi kopiju samo onih stranica baze koje su izmenjene od poslednjeg punog backup-a tih datoteka. Kao i kod diferencijalnog backup-a cele baze, ovaj pristup može smanjiti broj promena koje je potrebno zabeležiti za oporavak datoteka. U poređenju s diferencijalnim backup-om baze, prednost diferencijalnog backup-a datoteka je kraće vreme izvođenja i mogućnost fokusiranja na promene u specifičnim datotekama.

Delimična rezervna kopija datoteka, kao i potpuna i diferencijalna, omogućava kreiranje rezervnih kopija samo određenih delova baze podataka, bez potrebe za kopiranjem celokupne baze. Međutim, ova vrsta kopije ne obuhvata grupe fajlova koje su označene kao samo za čitanje. Ova opcija je korisna u situacijama kada baza sadrži grupe fajlova koje su nepromenjene i ne treba ih kopirati pri svakoj rezervnoj kopiji.

Delimična rezervna kopija omogućava kreiranje rezervnih kopija samo onih delova baze podataka koji su aktivni i koji se menjaju, dok se preskoče grupe fajlova koje su označene kao samo za čitanje. Ova metoda je posebno korisna za velike baze podataka koje sadrže obimne grupe fajlova koje se retko ažuriraju, kao što su arhivski podaci. Na taj način, vreme i resursi potrebni za izradu rezervne kopije se značajno smanjuju.

Rezervne kopije datoteka mozemo kreirati pomoću *SQL Server Management Studio*-a, izborom „*Files and filegroups*“ komponente, a slicno i za baze podataka samo umesto „*Files and filegroups“ biramo* „*Database*“ (*Slika 14.*).



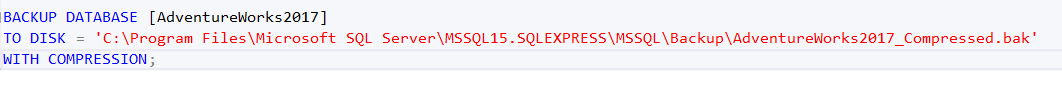
Slika 14. Kreiranje file backup-a pomoći SQL Server Management Studio-a

# **Kompresija i enkripcija rezervnih kopija**

Kompresija rezervnih kopija je tehnika koja značajno smanjuje veličinu rezervne kopije baze podataka, čime se štedi prostor na disku i skraćuje vreme potrebno za kreiranje i obnavljanje rezervnih kopija. SQL Server pruža podršku za kompresiju rezervnih kopija od verzije SQL Server 2008. Ova tehnika koristi algoritme kompresije koji smanjuju količinu podataka koji se skladišti u rezervnoj kopiji. Algoritmi kompresije, kao što su Huffmanovo kodiranje ili Lempel-Ziv, omogućavaju smanjenje prostora potrebnog za skladištenje.

Kompresija može doneti koristi u smislu smanjenja vremena potrebnog za kreiranje rezervnih kopija i smanjenja prostora koji one zauzimaju, ali može povećati opterećenje CPU-a tokom procesa kreiranja rezervnih kopija. Preporučuje se da se uticaj na performanse sistema pažljivo testira pre implementacije. Kompresija se može omogućiti putem SQL Server Management Studio (SSMS) ili T-SQL komandi koristeći opciju WITH COMPRESSION. Na primer, komanda za uključivanje kompresije može izgledati ovako:

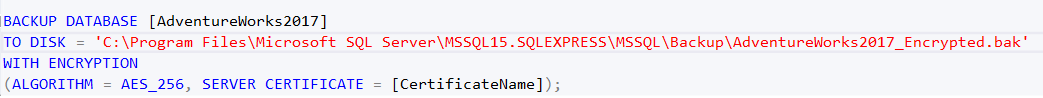
Naredba kreiranja kompresovane rezervne kopije datoteka:



Slika 15. Naredba kreiranja kompresovane rezervne kopije podataka

**Prednosti kompresije** uključuju smanjenje prostora potrebnog za skladištenje rezervnih kopija i skraćivanje vremena za njihovo kopiranje ili slanje preko mreže. Međutim, **nedostaci** uključuju povećano CPU opterećenje tokom procesa kompresije i moguće uticaje na performanse sistema, naročito kod velikih baza podataka.  
  
 Enkripcija rezervnih kopija pruža dodatni nivo zaštite podataka tako što ih štiti od neovlašćenog pristupa. SQL Server omogućava enkripciju rezervnih kopija od verzije SQL Server 2014. Enkripcija se koristi za zaštitu podataka u rezervnim kopijama korišćenjem algoritama kao što su AES (Advanced Encryption Standard) i Triple DES (Data Encryption Standard). Ključ enkripcije može biti simetričan, gde se koristi isti ključ za enkripciju i dekripciju, ili asimetričan, gde se koriste različiti ključevi za ove procese.

Enkripcija se može omogućiti prilikom kreiranja rezervnih kopija koristeći SQL Server Management Studio (SSMS) ili T-SQL komande. Ključ enkripcije mora biti pažljivo upravljan i zaštićen. Komanda za omogućavanje enkripcije izgleda ovako:



Slika 16. Naredba za kreiranje enkriptovane rezervne kopije podataka

Prednosti enkripcije uključuju zaštitu podataka i usklađenost sa regulativama koje zahtevaju sigurnost podataka. S druge strane, nedostaci obuhvataju povećano CPU opterećenje tokom enkripcije i potrebu za pažljivim upravljanjem ključevima. Gubitak ključeva može onemogućiti pristup podacima iz rezervnih kopija.

Korišćenje kompresije i enkripcije rezervnih kopija u SQL Serveru poboljšava performanse i sigurnost sistema. Kompresija smanjuje prostor potreban za skladištenje i ubrzava proces kreiranja rezervnih kopija, dok enkripcija osigurava podatke i štiti ih od neovlašćenog pristupa. Implementacija ovih tehnika treba da se izvrši uz razmatranje njihovog uticaja na performanse sistema i potrebu za upravljanjem ključevima.

# **Restore baze podataka**

Oporavak baze podataka, odnosno restore baze podataka, predstavlja način na koji se čuvaju i održavaju transakcije, kao i način na koji se vrši oporavak i vraćanje podataka u slučaju greske. Kod SQL Server postoje 3 modela oporavka:

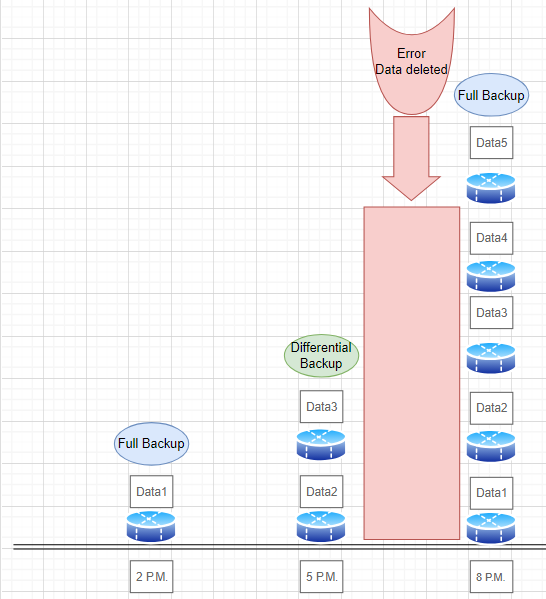
1. Jednostavan model oporavka (eng. Simple recovery model) – dozvoljeno je izvršiti backup potpune i diferencijalne rezervne kopije baze podataka, dok nije dozvoljeno izvršiti transaction log backup. Ovako nesto je pogodno za jednostavne baze sa manjim zahtevima.
2. Potpuni model oporavka (eng. Full recovery model) – za razliku od prvog modela, u ovom modelu se sve transakcije čuvaju u transaction log backup-u. Prilikom opravka podataka kombinuju su potpuni, diferencijalni i transaction-i log backup-a.
3. Bulk-logged model oporavka – ima slične karakteristike kao i prethodni, ali sa nekim ograničenjima. Namenjen je za velike operacije upisivanja/ažuriranja kako bi se smanjio uticaj na performanse. Kod oporavka podataka kombinuje se potpuni, diferencijalni i transaction-i log backup-a.

Kako bi se izvršio oporavak baze podataka nakon neke greske ili neuspeha, neophodno je vratiti skup rezervnih kopija u ispravnom i smislenom redosledu. SQL Server podržava nekoliko tipova vraćanja podataka:

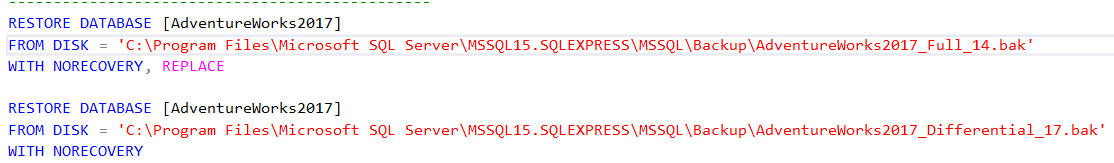
1. Vraćanje celokupne baze podataka (eng. complete database restore) – vrši obnavljanje cele baze podataka iz rezervnih kopija, uključujući sve fajlove i strukture podataka koji čine bazu podataka. Predstavlja najsveobuhnatiji oblik vraćanja podataka i koristi se kada je potrebno napraviti restore cele baze nakon greške ili neuspeha.
2. Vraćanje datoteke sa podacima (eng. file restore) – vrši obnavljanje pojedinačnih fajlova ili grupe fajlova sa podacima iz rezervnih kopija. Koristan u slucaju kada je određeni fajl sa podacima oštećen ili izgubljen, a ostatak baze podataka je sačuvan. Na ovaj način možemo obnoviti samo oštećeni deo podataka.
3. Vraćanje stranice sa podacima (eng. page restore) – vrši obnavljanje pojedinačne stranice sa podacima iz rezervnih kopija. Ovakva vrsta vraćanja podataka se koristi u retkim slučajevima kada je samo mali deo podataka oštećen ili izgubljen. Obnavljanje stranica sa podacima se obično vrši automatski tokom procesa oporavka. Postoji mogućnost i ručnog obnavljanja ukoliko je to potrebno.

Primeri upotrebe *RESTORE* naredbe kod SQL Serveru:

1. *Primer jednostavnog oporavka* (*Slika 15.*): Potpuna rezervna kopija baze podataka je napravljena u 14h i 20h, a diferencijalna rezervna kopija u 17h. Pretpostavimo da se u 18h dogodio problem i da su neki ključni podaci obrisani. U ovom slučaju, najbolje je vratiti potpunu rezervnu kopiju (iz 10h), a zatim diferencijalnu rezervnu kopiju (iz 15h) (*Slika 17.*). Problem u ovom slučaju oporavka je to da će sve promene koje su nastale između 15h i 22h biti izgubljene.

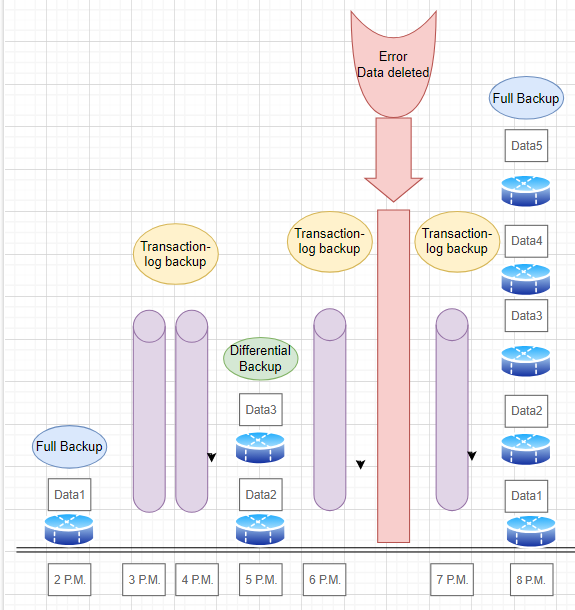


Slika 17. Raspored pravljenja backup-a i nastanak problema

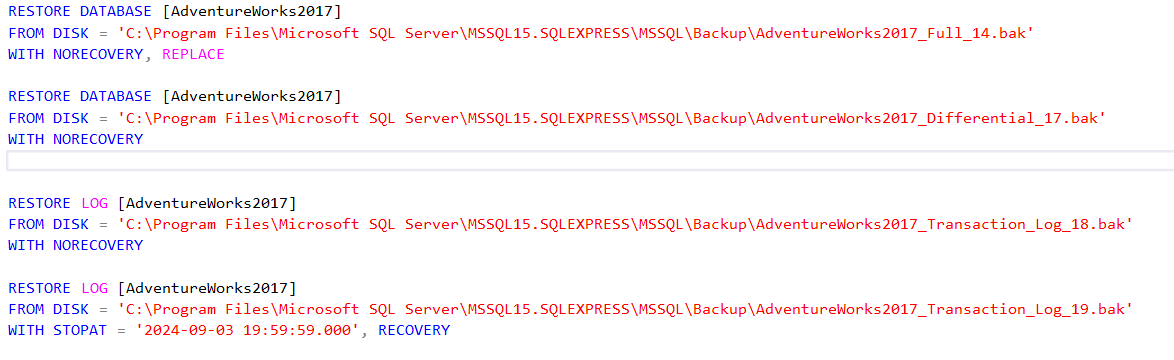


Slika 18. problema Naredbe vraćanja podataka prilikom jednostavnog oporavka

1. *Primer potpunog oporavka* Slika 19*.*):Potpuna rezervna kopija baze podataka je napravljena u 14h i 20h, diferencijalna rezervna kopija u 17h, a transaction log backup u 15, 16, 18 i 19h. Problem je nastao u 19h, što znači da će sve promene između 19h i 20h biti izgubljene. Najbolji način za oporavak podataka u ovom slučaju jeste vraćanje potpune rezervne kopije (iz 10h), diferencijalne kopije (iz 15h) i rezervne kopije evidencije izvršenih transakcija (18h i 19h) (*Slika 19.*).

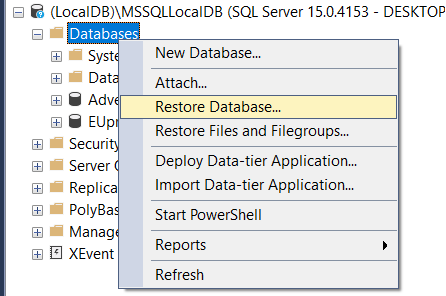
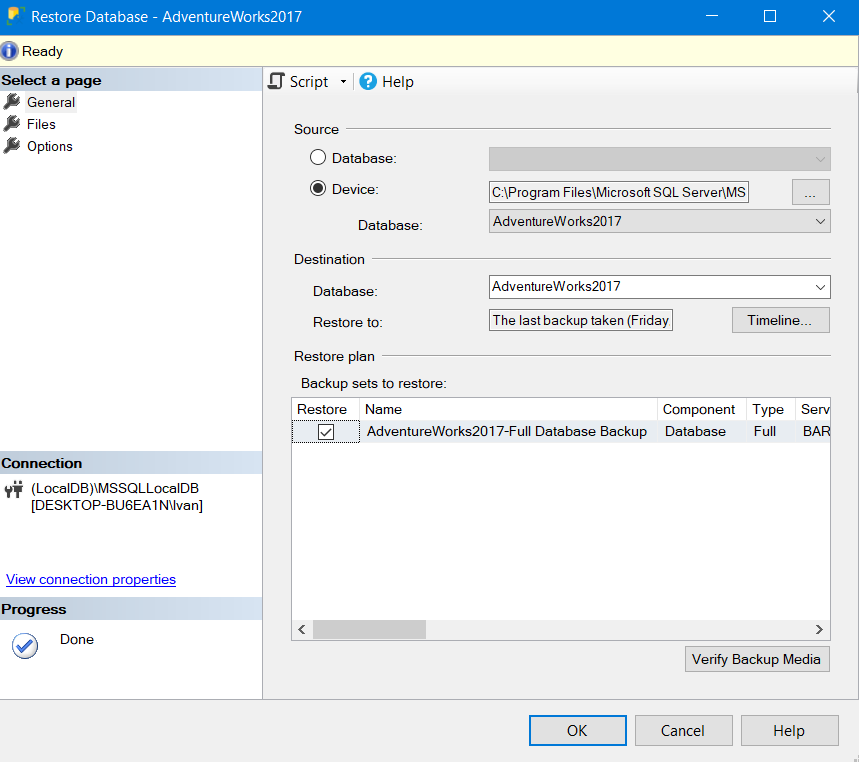


Slika 19. Raspored pravljenja backup-a i nastanak problema



Slika 20. Naredbe vraćanja podataka prilikom potpunog oporavka baze podataka

Restore je moguće izvršiti i pomoću *SQL Server Management Studio*-a   
(*Database Tasks Restore*), izborom vrste restore procesa, zatim vrste backup-a koji koristimo za vraćanje podataka (*Slika 19,20,21*).

Slike 21,22,23 – Proces restore podataka pomoću SQL Server Management Studio-a

# **Planovi za oporavak od katastrofe**

Planovi za oporavak od katastrofe (Disaster Recovery Plans - DRP) predstavljaju ključnu komponentu strategije zaštite podataka i kontinuiteta poslovanja u svakoj organizaciji. Oni su dizajnirani kako bi se osiguralo da se sistem može brzo i efikasno vratiti u funkciju nakon ozbiljnih kvarova, prirodnih nepogoda, ljudskih grešaka ili drugih nepredviđenih događaja koji mogu ugroziti integritet i dostupnost podataka.

Prvi korak u izradi plana za oporavak od katastrofe jeste procena rizika. Ovaj korak uključuje identifikaciju potencijalnih pretnji koje mogu uticati na rad sistema. To može obuhvatati prirodne katastrofe kao što su poplave i požari, tehničke kvarove kao što su problemi sa hardverom ili mrežom, ljudske greške i sajber pretnje poput napada ransomware-om. Analizom ovih rizika, organizacija može bolje razumeti koje vrste katastrofa su najverovatnije i kako mogu da ih ublaže.

Nakon procene rizika, važno je postaviti ciljeve oporavka, koji uključuju ciljeve oporavka podataka (RPO - Recovery Point Objective) i ciljeve vremena oporavka (RTO - Recovery Time Objective). RPO definiše maksimalno prihvatljiv period gubitka podataka koji se može dogoditi tokom oporavka. Na primer, ako je RPO postavljen na jedan sat, organizacija mora osigurati da podaci ne budu izgubljeni za više od jednog sata. S druge strane, RTO predstavlja maksimalno vreme koje je prihvatljivo za vraćanje sistema u funkciju nakon katastrofe. Ako je RTO četiri sata, sistem mora biti operativan u roku od četiri sata nakon incidenta.

Implementacija dobro razrađenog plana za oporavak od katastrofe omogućava organizacijama da minimiziraju prekide u radu, zaštite svoje podatke i obezbede brz povratak u normalan rad nakon ozbiljnih problema ili katastrofa.

# **Zaključak**

Backup i restore procesi kod sistema za upravljanje bazama podataka (DBMS) su ključni prilikom zaštite podataka i omogućavaju oporavak u slučaju gubitka ili oštećenja podataka. Backup proces omogućava sigurnost podataka tako što pravi kopije originalnih podataka i čuva ih na sigurnom mestu. U slučaju gubitka ili oštećenja originalnih podataka usled hardverskih kvarova, grešaka korisnika, zlonamernih napada ili prirodnih katastrofa, backup podaci se mogu koristiti za obnovu izgubljenih ili oštećenih podataka. Ako dođe do nenamernog brisanja, izmene ili oštećenja podataka, backup omogućava njihovo vraćanje na prethodno stanje, čime se čuva njihova tačnost i pouzdanost.

Postoji nekoliko osnovnih tipova rezervnih kopija koje podržava Microsoft SQL Server, uključujući potpuni (eng. full backup) koji čuva kompletnu kopiju cele baze podataka, diferencijalni (eng. differential backup) koji čuva samo promene od poslednje potpune rezervne kopije i transaction log backup koji čuva evidencije izvršenih transakcija. Osnovni tipovi rezervnih kopija se mogu kombinovati u cilju pronalaženja što bolje strategije za backup i restore. Rezervne kopije se ne moraju uvek vršiti nad bazom podataka, već je moguće i izraditi rezervnu kopiju datoteka.

Restore proces podrazumeva vraćanje podataka iz rezervne kopije u bazu podataka i pri- tom je važno da se poštuje pravilan redosled operacija i da se koristi odgovarajuća strategija u skladu sa zahtevima oporavka podataka. Model oporavka baze podataka takođe utiče na način i vrstu restore procesa koji se primenjuje.

Da bi se obezbedio uspešan backup i restore proces, neophodno je imati dobro osmišljenu strategiju. Strategija uključuje izbor vrste backup-a, frekvenciju izvršavanja backup-a, njegovo testiranje i pronalaženje sigurne lokacije za čuvanje rezervnih kopija. Testiranje strategije vraćanja podataka je ključni korak kako biste bili sigurni da je vaša strategija efikasna i da se može primeniti u slučaju nastanka problema u bazi podataka. Testiranje uključuje izbor određenih backup fajlova, njihovo vraćanje i proveru da li je baza podataka uspešno obnovljena i funkcionalna. Testiranje takođe omogućava identifikovanje eventualnih problema ili nedostataka u strategiji i omogućava vam da ih rešite pre nego što dođe do problema u realnoj situaciji. Takođe je važno odrediti raspored izvršavanja backup-a, kako bi se osiguralo da se podaci redovno čuvaju. Pravilno upravljanje ovim procesima omogućava se zaštita podataka od gubitka ili oštećenja, i brz oporavak u slučaju neuspeha.

Procesi pravljenja rezervnih kopija i vraćanja podataka u sistemima za upravljanje bazama podataka (DBMS) ključni su za očuvanje podataka i omogućavaju oporavak u slučaju njihovog gubitka ili oštećenja. Pravilno vršenje rezervnih kopija osigurava podatke tako što stvara kopije originalnih informacija i smešta ih na sigurno mesto. Ako se dogodi gubitak ili oštećenje originalnih podataka zbog hardverskih problema, ljudskih grešaka, napada zlonamernog softvera ili prirodnih nepogoda, ove kopije mogu se koristiti za obnavljanje izgubljenih ili oštećenih informacija. Takođe, ako dođe do nenamernih izmena, brisanja ili oštećenja podataka, rezervne kopije omogućavaju vraćanje podataka u prethodno stanje, čime se osigurava njihova tačnost i pouzdanost.

Microsoft SQL Server nudi različite vrste rezervnih kopija, uključujući potpune rezervne kopije koje obuhvataju celu bazu podataka, diferencijalne rezervne kopije koje beleže samo promene od poslednje potpune kopije, i transaction log kopije koje sadrže evidenciju izvršenih transakcija. Ove vrste rezervnih kopija mogu se kombinovati kako bi se razvila optimalna strategija za pravljenje i vraćanje rezervnih kopija. Takođe, rezervne kopije se ne moraju ograničiti samo na bazu podataka, već se mogu napraviti i kopije datoteka.

Proces vraćanja podataka uključuje obnavljanje informacija iz rezervnih kopija u bazu podataka, pri čemu je važno da se operacije izvrše u ispravnom redosledu i da se koristi odgovarajuća strategija u skladu sa zahtevima oporavka. Model oporavka baze podataka takođe utiče na način i vrstu procesa vraćanja koji će se primeniti.

Za uspešno obavljanje backup i restore procesa, potrebno je imati jasno definisanu strategiju. Ova strategija obuhvata izbor vrste rezervnih kopija, učestalost njihovog pravljenja, testiranje i odabir sigurnih lokacija za skladištenje rezervnih kopija. Testiranje strategije vraćanja podataka je ključan korak kako biste osigurali da je vaša strategija funkcionalna i da može biti primenjena u slučaju problema sa bazom podataka. Testiranje podrazumeva odabir određenih rezervnih kopija, njihovo vraćanje i proveru da li je baza podataka uspešno obnovljena i operativna. Takođe, testiranje pomaže u prepoznavanju potencijalnih problema ili nedostataka u strategiji, omogućavajući vam da ih rešite pre nego što nastupe u stvarnim uslovima. Takođe, važno je odrediti raspored pravljenja rezervnih kopija kako bi se osiguralo da se podaci redovno čuvaju. Efikasno upravljanje ovim procesima omogućava zaštitu podataka od gubitka ili oštećenja i brzi oporavak u slučaju neuspeha.

# **Literatur****a**

Database backup and restore, <https://learn.microsoft.com/en-us/sql/relational-databases/backup-restore/quickstart-backup-restore-database?view=sql-server-ver16&tabs=ssms>

Full backup of database, <https://sqlbak.com/academy/full-backup/>

Differential bacup of database, <https://sqlbak.com/academy/differential-backup/>

Transaction log backup of database, <https://sqlbak.com/academy/transaction-log-backup/>   
Tail only backup of database, <https://sqlbak.com/blog/tail-log-backups/>

Copy only backup of database, <https://sqlbak.com/academy/copy_only_backup/>

Data extents, <https://sqlbak.com/academy/differential-backup/data-extents/>

SQL backup types , <https://medium.com/coinmonks/three-top-sql-server-backup-types-fbd253fc09f3>   
Encrypted backup of database, <https://learn.microsoft.com/en-us/sql/relational-databases/backup-restore/backup-encryption?view=sql-server-ver16>   
File group backups ,<https://sqlbak.com/academy/file-backups/>

Compression of backup database, <https://learn.microsoft.com/en-us/sql/relational-databases/backup-restore/backup-compression-sql-server?view=sql-server-ver16>   
Disaster recovery plan, <https://www.red-gate.com/simple-talk/databases/sql-server/learn/sql-server-high-availability-and-disaster-recovery-plan/>