[Integrita databáze](https://cs.wikipedia.org/wiki/Integrita_datab%C3%A1ze" \o "Integrita databáze) znamená, že data v ní uložená jsou konzistentní vůči definovaným pravidlům. Lze zadávat pouze data, která vyhovují předem definovaným kritériím (např. musí respektovat datový typ nastavený pro daný sloupec tabulky, či další omezení hodnot přípustných pro daný sloupec).

 K zajištění integrity slouží [integritní omezení](https://cs.wikipedia.org/wiki/Integritn%C3%AD_omezen%C3%AD). Jedná se o nástroje, které zabrání vložení nesprávných dat či ztrátě nebo poškození stávajících záznamů v průběhu práce s databází. Typicky je možné zajistit mazání dat, která již ztratila svůj význam (kupř. smažeme-li uživatele z tabulky uživatelů, odstraní se i na něj navázané záznamy v ostatních databázových tabulkách).

Integritní omezení jsou součástí definice databáze, za jejich splnění odpovídá SŘBD.

**Druhy integritních omezení**

·        *Entitní integritní omezení* – někdy je mylně považováno za pouhé omezení primárního klíče tabulky (jež má zamezit uložení dat, která neobsahují všechna pole sdružená do klíče, nebo data, jež by v těchto polích byla stejná jako v nějakém jiném, již zapsaném, řádku tabulky). Ve skutečnosti jde o to, aby o jedné entitě nebylo možno do databáze vložit duplicitní záznamy; jedná se tedy o zajištění unikátnosti skutečných identifikátorů reálných objektů (což například umělé automaticky generované ID nezajistí).

·        *Doménová integritní omezení* – zajišťují dodržování datových typů/domén definovaných sloupcům databázové tabulky.

·        *Aktivní referenční integrita* – definují činnosti, které databázový systém provede, pokud jsou porušena některá pravidla.

·        *Referenční integritní omezení* – zabývají se vztahy dvou tabulek, kde jejich relace je určena vazbou primárního a cizího klíče. Jedná se o požadavek, aby pole záznamu, jenž má obsahovat odkaz na jiný záznam někde v databázi, takový odkazový záznam skutečně existoval, tedy aby takový záznam nevedl do prázdna a nejednalo se o tzv. databázového sirotka.

**Dodržování integritních omezení**

1.     umístění jednoduchých mechanismů pro dodržování integritních omezení na straně databázového serveru. Jedná se o nejlepší způsob z hlediska ochrany dat. Uživateli však obvykle přináší delší odezvu systému a nelze vždy zajistit jejich přenositelnost na jiný databázový systém.

2.     umístění ochranných mechanismů na straně klienta. Pro komfort a nezávislost na databázovém systému je nejlepší volbou. Nutnost kontrolních mechanismů pro každou operaci může způsobit chyby u aplikací a v případě většího počtu aplikací je potřeba je opravit na více místech.

3.     samostatné programové moduly na straně serveru. V moderních databázových systémech jsou pro tento účel implementovány tzv. [triggery](https://cs.wikipedia.org/wiki/Trigger" \o "Trigger). Jedná se o samostatné procedury, které lze spouštět automatizovaně před a po operacích manipulujících s daty. Tento způsob umožňuje implementaci i složitých integritních omezení. Nevýhody opět přináší provádění na serveru, i velmi omezená možnost přenesení na jiný databázový systém.

Možná je i kombinace předchozích variant v závislosti na konkrétních podmínkách.

Kontroly integritních omezení se zpravidla provádějí po každé provedené operaci, což snižuje nároky na server. Není nutno nijak zaznamenávat, které kontroly mají být provedeny později. Složitější integritní omezení však vždy nelze takto ověřit, proto je možné kontrolovat dodržení pravidel až po dokončení celé transakce.