

Principali informazioni sull'insegnamento	
Titolo insegnamento	Progettazione di Basi di Dati
Corso di studio	Informatica e Tecnologie per la Produzione del Software – L3I
Crediti formativi	9
Denominazione inglese	Data Base Design
Obbligo di frequenza	no
Lingua di erogazione	Italiano

Docente responsabile	Nome Cognome	Indirizzo Mail
	Filippo Tangorra	filippo.tangorra@uniba.it

Dettaglio credi formativi	Ambito disciplinare	SSD	Crediti
	Informatico- L3I	INF/01 -Informatica	9

Modalità di erogazione	
Periodo di erogazione	Primo semestre
Anno di corso	Secondo anno
Modalità di erogazione	Lezioni frontali (7 cfu) Esercitazioni in aula e laboratorio (2 cfu)

Organizzazione della didattica	
Ore totali	225
Ore di corso	86
Ore di studio individuale	139

Calendario	
Inizio attività didattiche	24 settembre 2018
Fine attività didattiche	11 gennaio 2019

Syllabus	
Prerequisiti	
Risultati di apprendimento previsti (declinare rispetto ai Descrittori di Dublino) (si raccomanda che siano coerenti con i risultati di apprendimento del CdS, compreso i risultati di apprendimento trasversali)	<ul style="list-style-type: none"> • <i>Conoscenza e capacità di comprensione</i> lo studente acquisirà le principali conoscenze riguardanti i modelli per basi di dati, i linguaggi di interrogazione e le metodologie di progetto comunemente utilizzati ai livelli concettuale e logico della progettazione di basi di dati. Lo studente possiederà le conoscenze formali che gli consentiranno di comprendere la struttura delle basi di dati rappresentate con il modello Entità Relazione (livello concettuale), con il modello relazionale (livello logico) e le interrogazioni ai dati espresse il linguaggio SQL. • <i>Conoscenza e capacità di comprensione applicate</i> lo studente sarà in grado di applicare le conoscenze acquisite con l'obiettivo di progettare, realizzare e gestire una base di dati corretta. Tali conoscenze gli consentiranno di produrre

	<p>schemi concettuali ER di basi di dati, di tradurli in schemi relazionali, e di implementarli in strumenti software tramite l'ausilio di sistemi quali MySQL.</p> <ul style="list-style-type: none"> • <i>Autonomia di giudizio</i> lo studente sarà in grado di valutare la correttezza e l'efficienza di una base di dati progettata secondo il modello previsto per ciascun livello di progettazione, e di operare le opportune scelte per garantire la qualità della base di dati. • <i>Abilità comunicative</i> lo studente sarà in grado di illustrare (anche in forma scritta) in modo formale, corretto e critico le soluzioni adottate nella progettazione della base di dati e le proprietà che essa deve soddisfare in termini di integrità e qualità di dati attraverso una documentazione appropriata. Egli saprà mostrare le abilità pratiche nella formulazione delle interrogazioni per la gestione dei dati. Il raggiungimento di un adeguato livello di abilità comunicativa viene valutato nell'esame finale di profitto. • <i>Capacità di apprendere</i> lo studente apprenderà la capacità di progettare e implementare, in autonomia, basi di dati a partire dai requisiti e dalle specifiche descrittive che descrivono la tipologia di contenuti ed operazioni necessarie alla gestione dei dati di interesse per l'applicazione. Le competenze acquisite gli consentiranno di capire e affrontare nuovi modelli, metodologie di progettazione, e ambienti software anche diversi da quelli proposti nel corso. Sarà anche in grado di inserirsi in gruppi di lavoro per la realizzazione di basi di dati di grandi dimensioni di supporto a Sistemi Informativi complessi.
<p>Contenuti di insegnamento</p>	<p><i>Concetti introduttivi</i> Cenni storici sulla evoluzione dei sistemi di gestione dei dati. Architettura a tre livelli ANSI/Sparc; indipendenza logico/fisica dei dati; I sistemi di gestione di basi di dati (DBMS), Modelli di basi di dati.</p> <p><i>Modello relazionale (livello logico)</i> Definizione e vincoli (sul dominio, di chiave, sui valori nulli. Vincoli di integrità della entità, integrità referenziale, chiavi esterne). Linguaggi relazionali. Algebra relazionale (unione, intersezione, differenza, complemento, complemento attivo, ridenominazione, selezione, proiezione, prodotto cartesiano, theta join, equi-join, join naturale, semi-join, join esterni, divisione). Dipendenze funzionali e loro proprietà. Regole di Armstrong. Chiusura di attributi. Equivalenza e minimalità di insiemi di dipendenze funzionali. Normalizzazione. Forme normali (1-2-3-NF e BCNF) e loro forme generali. Decomposizione con/senza perdita.</p>

	<p>Decomposizione con/senza conservazione delle dipendenze. Algoritmi di normalizzazione per analisi e per sintesi. Laboratorio: SQL-2 (linguaggio di definizione e manipolazione dei dati). Esercitazioni di laboratorio con MySQL.</p> <p><i>Modello Entità-Relazione (ER) (livello concettuale)</i> Tipi di entità, insiemi di entità, attributi e chiavi, tipi di associazioni, ruoli e vincoli strutturali, tipi di associazione di grado maggiore di due, tipi di entità debole, diagrammi ER. Estensione del modello Entità-Relazione (EER): sottoclassi, superclassi ed ereditarietà; specializzazione e generalizzazione diagrammi EER. Il dizionario dei dati. Vincoli di integrità e regole aziendali. Strategie di progetto di schemi concettuali.</p> <p><i>Progettazione logica.</i> Passi di progettazione logica e traduzione di schemi concettuali in schemi logici.</p>
--	---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

Programma	
Testi di riferimento	<p>P. Atzeni et al., Basi di Dati. Modelli e linguaggi di interrogazione, III edizione, McGraw-Hill Italia 2009.</p> <p>R.A. Elmasri, S.B. Navathe, Sistemi di basi di dati: fondamenti, Pearson - A.Wesley Italia, 5a – 6a Edizione, 2007-2010.</p> <p>L. Welling, L. Thomson MySQL Tutorial MySQLPress, Pearson Educ. Italia, 2004.</p>
Note ai testi di riferimento	Materiale didattico integrativo sarà fornito su piattaforma e-learning ADA.
Metodi didattici	Lezioni frontali ed esercitazioni. Esercitazioni pratiche in laboratorio sull'utilizzo di Mysql.
Metodi di valutazione (indicare almeno la tipologia scritto, orale, altro)	Una prova pratica di laboratorio e una prova scritta.
Criteri di valutazione (per ogni risultato di apprendimento atteso su indicato, descrivere cosa ci si aspetta lo studente conosca o sia in grado di fare e a quale livello al fine di dimostrare che un risultato di apprendimento è stato raggiunto e a quale livello)	<p>La verifica dell'apprendimento avverrà già in itinere, durante l'espletamento del corso, con valore esonerante dalla corrispondente parte di esame di profitto in caso di valutazione positiva.</p> <p>L'esame si compone di una prova di laboratorio e una prova scritta. La prova di laboratorio valuterà le capacità dello studente di creare ed interrogare una base di dati relazionale in MySQL, e sarà propedeutica alla prova scritta. La prova scritta di esame valuterà la capacità dello studente sia di illustrare i concetti fondamentali introdotti nel corso, che di risolvere casi pratici di progettazione concettuale e logica di una base di dati, sua interrogazione in Algebra relazionale ed ottimizzazione tramite le forme di normalizzazione.</p> <p>La prova di laboratorio e quella scritta debbono essere sostenute nello stesso appello.</p>
Altro	