



Principali informazioni sull'insegnamento

Denominazione dell'insegnamento	Progettazione di Basi di Dati	
Corso di studio	Informatica e Tecnologie per la Produzione del Software	
Anno Accademico	2024/25	
Crediti formativi universitari (CFU) / European Credit Transfer and Accumulation System (ECTS)	9 CFU	
Settore Scientifico Disciplinare	INF/01	
Lingua di erogazione	Italiano	
Anno di corso	Primo	
Periodo di erogazione	1^ semestre, le date esatte sono riportate nel manifesto/regolamento	
Obbligo di frequenza	No, ma la frequenza è fortemente raccomandata	
Sito web del corso di studio	https://www.uniba.it/it/ricerca/dipartimenti/informatica/didattica/corsi-di-laurea/informatica-tps-270/laurea-triennale-in-informatica-e-tecnologie-per-la-produzione-del-software-d.m.-270	

Docente/i	<i>Nel caso in cui il docente sia più d'uno, aggiungere un box intero per l'altro docente</i>
Nome e cognome	Michelangelo Ceci
Indirizzo mail	michelangelo.ceci@uniba.it
Telefono	+39 080 5442285
Sede	Dipartimento di Informatica, Via Orabona 4, 70125, Bari. Stanza n.523, 5^ piano
Sede virtuale	Piattaforma e-learning UNIBA - https://elearning.uniba.it/
Sito web del docente	http://www.di.uniba.it/~ceci/
Ricevimento (giorni, orari e modalità, es. su appuntamento)	In presenza presso lo studio del docente: Giovedì, 9:00-11:00, preferibilmente su appuntamento

Syllabus	
Obiettivi formativi	Lo studente apprenderà a progettare concettualmente e logicamente una base di dati, analizzando i requisiti raccolti e applicando determinate metodologie per garantire



	alcune proprietà desiderate nella base di dati realizzata. Lo studente, inoltre, acquisirà la capacità di interrogare efficientemente le basi di dati. Lo studente apprenderà ad implementare praticamente una base di dati in uno specifico DBMS. Il DBMS considerato durante il corso sarà MySQL.
Prerequisiti	Sebbene non sia un forte requisito, può essere di aiuto la conoscenza di nozioni su algebra, in particolare per la costruzione di espressioni in algebra relazionale. La conoscenza di linguaggi formali può aiutare nella comprensione e nella scrittura di espressioni ben formate in SQL. La lingua inglese aiuta a comprendere i termini utilizzati nei vari linguaggi di interrogazione.
Contenuti di insegnamento (Programma)	1. Introduzione ai sistemi per basi di dati (10h) I sistemi organizzativi. I sistemi informativi per la produzione. I sistemi informatici: componenti ed evoluzioni. Requisiti di un sistema informatico complesso. Basi di dati e sistemi di gestione di basi di dati. I modelli dei dati. Livelli di astrazione nei DBMS. Linguaggi per data base. Interfacce per DBMS. Utenti della base di dati. Controllo della base di dati: integrità, affidabilità, sicurezza. Classificazione dei DBMS. I moduli di un DBMS. Vantaggi e problemi nell'uso dei DBMS. I modelli dei dati: gerarchico, reticolare, relazionale. 2. Basi di dati relazionali (6h): modello. Relazioni e tabelle. Relazioni con attributi. Relazioni e basi di dati. Informazione incompleta e valori nulli. Vincoli d'integrità. Vincoli di tupla. Chiavi. Chiavi e valori nulli. Vincoli di integrità referenziale. 3. Basi di dati relazionali: linguaggi (20h). Algebra relazionale. Operatori primitivi (unione, differenza, ridenominazione, selezione, proiezione, prodotto cartesiano), operatori derivati (intersezione, divisione, giunzione, giunzione naturale, giunzione esterna, semi-giunzione), altri operatori (complemento, funzioni di aggregazione, chiusura transitiva). Proprietà algebriche degli operatori relazionali. Algebra e calcolo con valori nulli. Calcolo relazionale. Calcolo relazionale su domini, calcolo su tuple con dichiarazioni di range. 4. SQL. Storia ed evoluzione. Interrogazioni semplici in SQL. Interrogazioni di tipo insiemistico. Interrogazioni con raggruppamento. SQL per definire e amministrare basi di dati: creazione di una base di dati, creazione di tabelle, definizione di domini, inserimento/cancellazione/modifica di tuple, vincoli d'integrità (intra-relazionali e inter-relazionali), modifica degli schemi, viste logiche, asserzioni, aspetti fisici (parametri fisici, definizioni di indici), cataloghi relazionali, controllo dell'accesso, strumenti per l'amministrazione di basi di dati. SQL per programmare le applicazioni: linguaggi che ospitano l'SQL, linguaggi con interfaccia API, linguaggi integrati, la programmazione di transazioni (ripetizione esplicita delle transazioni, transazioni con livelli diversi di isolamento). 5. La progettazione di basi di dati (20h). Metodologie e modelli per il progetto. Il ciclo di vita dei sistemi informatici, una metodologia di progettazione per basi di dati, il modello entità-relazione (costrutti e documentazione degli schemi). La progettazione concettuale. La raccolta e l'analisi dei requisiti, i criteri generali di rappresentazione, le strategie di progetto (top-down, bottom-up, inside-out, ibrida), qualità di uno schema concettuale, una metodologia generale. La progettazione logica. Analisi delle prestazioni su schemi E-R, ristrutturazione di schemi E-R (analisi delle ridondanze, eliminazione delle gerarchie, partizionamento/accorpamento di concetti, scelta degli identificatori principali), traduzione verso il modello relazionale (entità e associazioni molti a molti, associazioni uno a molti, entità con identificatore esterno, associazioni uno a uno, rappresentazione grafica delle traduzioni, traduzione di schemi complessi, tabelle riassuntive). La normalizzazione. Ridondanze e anomalie, dipendenze funzionali, forma normale di Boyce e Codd, proprietà delle decomposizioni (decomposizione senza perdita, conservazione delle dipendenze, qualità delle decomposizioni), progettazione di basi di dati e normalizzazione (verifiche di normalizzazione su entità e associazioni, violazione di forme normali e ristrutturazione di schemi concettuali). Laboratorio (30h) 6. Il DBMS MySQL Caratteristiche del sistema. La creazione di un nuovo database. Tipi di dato per i campi di una tabella. Le relazioni (uno-a-uno, uno-a-molti, molti-a-molti). Integrità referenziale. Chiavi primarie e indici. Creazione di query di selezione (su singola tabella e su più tabelle), di comando e per parametri. Gli operatori e le espressioni in MySQL. Aggiornamento dei dati di tabella con le query. Query con funzioni di aggregazione. 6.



	Interfacciamento con JAVA. JDBC: introduzione, Librerie, Interrogazioni e operazioni di manipolazione di dati. Esercizi in laboratorio.			
7. Testi di riferimento	P. Atzeni, S. Ceri, P. Fraternali, S. Paraboschi, R. Torlone. Basi di dati. McGrawHill Education 5° Ed. 2018. ISBN 978-88-386-9445-5 Gli studenti che lo desiderano possono ottenere i testi in prestito dalla Biblioteca. Può convenire verificarne la disponibilità mediante il Sistema Bibliotecario di Ateneo https://opac.uniba.it/easyweb/w8018/index.php? e contattare la biblioteca per concordare il prestito. Gli studenti che lo desiderano possono ottenere i testi in prestito dalla Biblioteca. Può convenire verificarne la disponibilità mediante il Sistema Bibliotecario di Ateneo https://opac.uniba.it/easyweb/w8018/index.php? e contattare la biblioteca per concordare il prestito.			
Note ai testi di riferimento	Sul sito ADA sono disponibili: Le slide presentate a lezione, esempi di tracce di appelli precedenti, esempi di tracce della prova di laboratorio di appelli precedenti.			
Organizzazione della didattica				
Ore				
Totali	Didattica frontale	Laboratorio/esercitazione	Progetto	Studio individuale
225 ore	56 ore	30 ore	0 ore	139 ore
CFU/ETCS				
9 CFU	7 CFU	7 CFU	0 CFU	

Metodi didattici	
	Lezioni frontali sugli argomenti teorici del programma e su esempi pratici da sviluppare in aula e a casa. Attività in laboratorio per approfondire gli aspetti pratici e tecnologici. Esercitazioni periodiche (circa settimanali) permettono di consolidare e approfondire i concetti teorici. Inoltre, l'estensione per tutta la durata del corso di tali esercitazioni rappresenta un incentivo alla frequenza e all'autovalutazione del livello di apprendimento. Le esercitazioni sono pensate per attività da svolgere singolarmente

Risultati di apprendimento previsti	
Conoscenza e capacità di comprensione	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. Lo studente apprenderà a progettare concettualmente e logicamente una base di dati, analizzando i requisiti raccolti e applicando determinate



	metodologie per garantire alcune proprietà desiderate nella base di dati realizzata. Lo studente, inoltre, acquisirà la capacità di interrogare efficientemente le basi di dati.
Conoscenza e capacità di comprensione applicate	Lo studente acquisirà le conoscenze pratiche che gli consentiranno di applicare le conoscenze sulle basi di dati rappresentate con il modello Entità Relazione (livello concettuale), con il modello relazionale (livello logico) e le interrogazioni definite al fine di realizzare i modelli ER, logico e le interrogazioni in SQL. Lo studente apprenderà ad implementare praticamente una base di dati in uno specifico DBMS e ad interfacciarsi ad esso in Java.
Competenze trasversali	Autonomia di giudizio Lo studente acquisirà una autonomia di giudizio in quanto dovrà essere in grado di decidere le strategie da applicare in fase di progettazione concettuale e logica delle basi di dati. In particolare, sarà in grado di valutare aspetti relativi alla completezza, correttezza, minimalità degli schemi concettuali, nonché aspetti relativi all'efficienza nella progettazione di schemi logici. Abilità comunicative Lo studente sarà in grado di descrivere, attraverso documenti che seguono precise metodologie, le scelte intraprese durante le fasi di progettazione concettuale e logica. Ciò migliorerà le sue capacità di comunicazione nei confronti di possibili utilizzatori della base di dati. Capacità di apprendere in modo autonomo I concetti appresi non saranno solamente utili per gli argomenti esposti durante le lezioni frontali, ma risulteranno applicabili anche a contesti differenti (esempio, linguaggi diversi o DBMS diversi). Questo migliorerà la capacità di apprendimento dello studente, che sarà in grado di gestire situazioni analoghe, seppur diverse, rispetto a quelle esposte durante il corso.

Valutazione	
Modalità di verifica dell'apprendimento	Valutazione mediante una prova scritta ed una prova pratica. La prova scritta è svolta in modalità semistrutturata a risposta aperta. Essa è propedeutica alla prova pratica, che si svolge in laboratorio. Durante il corso si prevedono prove in itinere a carattere esonerante per la prova scritta. Il voto della prova scritta è mantenuto per l'intero Anno Accademico. La valutazione è in trentesimi per entrambe le prove e il voto finale è derivato applicando la media delle valutazioni delle prove. I risultati sono comunicati agli studenti tramite ADA per una visione preliminare e successivamente su Esse3.
Criteri di valutazione	



	La valutazione mira a verificare il raggiungimento di una buona capacità di problem solving nell'ambito delle basi di dati e dei sistemi informativi. In particolare, rispetto ai risultati di apprendimento attesi, si considerano i seguenti criteri: Conoscenza e capacità di comprensione · Lo studente dovrà dimostrare di conoscere e comprendere i principali modelli concettuali e logici dei dati, i linguaggi per basi di dati, le metodologie di progettazione delle basi di dati e le caratteristiche architetture dei sistemi per basi di dati. Tali aspetti sono oggetto di valutazione in entrambe le prove di esame. Conoscenza e capacità di comprensione applicate · Lo studente dovrà dimostrare di saper interpretare i bisogni informativi in un dato contesto applicativo, formalizzare tali bisogni per mezzo di specifiche, applicare i suddetti modelli e metodologie per la realizzazione di una base di dati che soddisfi le specifiche ed interrogare la base di dati così realizzata. Tali aspetti sono oggetto di valutazione in entrambe le prove di esame. Autonomia di giudizio · Lo studente dovrà dimostrare di saper formulare un proprio giudizio critico sulle diverse soluzioni progettuali e di saperlo sostenere all'interno di un gruppo di lavoro. Tale autonomia di giudizio è oggetto di valutazione nella prova scritta. Abilità comunicative · Lo studente dovrà dimostrare di saper comunicare la propria soluzione progettuale con appropriatezza di linguaggio, padronanza degli strumenti formali e gergo tecnico. Tali abilità comunicative sono oggetto di valutazione nella prova scritta. Capacità di apprendere · Lo studente deve dimostrare di aver acquisito la capacità di approfondire gli argomenti presentati a lezione, affrontando problemi di progettazione anche di difficoltà superiore a quelli proposti durante le esercitazioni. Tale capacità è oggetto di valutazione in entrambe le prove di esame.
Criteri di misurazione dell'apprendimento e di attribuzione del voto finale	L'apprendimento è valutato in base alla capacità di progettazione e implementazione di basi di dati. La valutazione è basata sulla padronanza e completezza degli argomenti previsti dal programma.
Altro	Si suggerisce agli studenti di affidarsi esclusivamente alle informazioni/comunicazioni fornite sui siti ufficiali del Dipartimento di Informatica, ovvero sui gruppi social solo se costituiti e amministrati esclusivamente dai docenti dei relativi insegnamenti: • https://www.uniba.it/it/ricerca/dipartimenti/informatica/didattica/corsi-di-laurea • https://www.uniba.it/it/ricerca/dipartimenti/informatica • https://elearning.uniba.it/ I programmi di tutti gli insegnamenti sono disponibili al seguente link: • https://elearning.uniba.it/ Le informazioni che tutti gli studenti dovrebbero conoscere sono scritte nei regolamenti didattici dei Corsi di Studi disponibili nel sito: • https://www.uniba.it/it/ricerca/dipartimenti/informatica/didattica/corsi-di-laurea



Si suggerisce agli studenti di diffidare delle informazioni e dei materiali circolanti su siti o gruppi social non ufficiali, poiché spesso sono risultati non affidabili, non corretti o incompleti. Per ogni dubbio, chiedere un incontro al docente secondo le modalità previste per il ricevimento.

Link al corso sulla piattaforma e-learning UNIBA:
<https://elearning.uniba.it/course/view.php?id=2801>



Main information on the course

Course name	Database Design	
Degree	Bachelor's Degree in Computer Science and Software Production Technologies	
Academic year	2024/25	
European Credit Transfer and Accumulation System (ECTS), in Italian Crediti Formativi Universitari (CFU)	9 CFU (each CFU corresponds to 25 hours (h) of student's time); CFU are of type T1, T2 or T3 T1 = 8 h lecture + 17 h individual study T2 = 15 h practice + 10 h individual study T3 = 25 h individual study	
Settore Scientifico Disciplinare	INF/01	
Course language	Italian	
Course year	Second	
Course period	First Semester - exact dates can be found in the didactic regulations	
Course attendance requirement	None, but it is highly recommended to attend classes	
Website of the Degree	https://www.uniba.it/it/ricerca/dipartimenti/informatica/didattica/corsi-di-laurea/informatica-tps-270/laurea-triennale-in-informatica-e-tecnologie-per-la-produzione-del-software-d.m.-270	

Teacher(s)

Name and Surname	Michelangelo Ceci
email	michelangelo.ceci@uniba.it
phone	+39 080 5442285
office	Dipartimento di Informatica, Via Orabona 4, 70125, Bari. Room n.510, 5th floor.
e-learning platform	Platform ADA - https://elearning.uniba.it/
Teacher's homepage	http://www.di.uniba.it/~ceci/
Office hours	Thursdays 15:00-17:00 - Student office hours can be scheduled via email

Syllabus

Course goals	The student will learn to conceptually and logically design a database, analyzing the gathered requirements and applying specific methodologies to ensure certain desired properties in the implemented database. Additionally, the student will acquire the ability to efficiently query databases. The student will also learn to practically implement a database in a specific Database Management System (DBMS). The DBMS covered in the course will be MySQL.
Prerequisites/requirements	While not a strict requirement, familiarity with algebraic concepts, particularly in constructing expressions in relational algebra, may be beneficial. Knowledge of formal languages can aid in understanding and writing well-formed SQL expressions. Proficiency in English helps in comprehending terms used in various query languages.
Course program	The course is structured as follows: Introduction to Database Systems (10h):



		Organizational systems. Information systems for production. Computer systems: components and evolution. Requirements of a complex computer system. Databases and Database Management Systems (DBMS). Data models. Abstraction levels in DBMS. Languages for databases. DBMS interfaces. Database users. Database control: integrity, reliability, security. Classification of DBMS. Modules of a DBMS. Advantages and issues in using DBMS. Data models: hierarchical, network, relational. Relational Databases (6h): Model. Relations and tables. Relations with attributes. Relations and databases. Incomplete information and null values. Integrity constraints. Tuple constraints. Keys. Keys and null values. Referential integrity constraints. Relational Databases: Languages (20h): Relational Algebra. Relational algebra operators. Algebraic properties of relational operators. Algebra and calculus with null values. Relational Calculus. Relational calculus on domains. Tuple calculus with range declarations. SQL. History and evolution. Simple SQL queries. Set-type queries. Grouping queries. SQL for defining and administering databases. SQL for programming applications. Transaction programming. Database Design (20h): Methodologies and models for design. Life cycle of computer systems. Design methodology for databases. Entity-Relationship model (constructs and schema documentation). Conceptual design. Requirements collection and analysis. General representation criteria. Project strategies (top-down, bottom-up, inside-out, hybrid). Quality of a conceptual schema. General methodology. Logical design. Performance analysis on E-R schemas. E-R schema restructuring. Translation to the relational model. Normalization. Boyce and Codd normal form. Properties of decompositions. Database design and normalization. Laboratory (30h): 5. MySQL DBMS: System characteristics. Creating a new database. Data types for table fields. Relationships (one-to-one, one-to-many, many-to-many). Referential integrity. Primary keys and indexes. Creating selection queries (on single and multiple tables), commands, and parameters. Operators and expressions in MySQL. Updating table data with queries. Queries with aggregation functions. Integration with JAVA: JDBC: introduction, libraries, queries, and data manipulation operations.		
Books of reference		P. Atzeni, S. Ceri, P. Fraternali, S. Paraboschi, R. Torlone. Basi di dati. McGrawHill Education 5° Ed. 2018. ISBN 978-88-386-9445-5 Students who wish to can borrow the texts from the Library. It may be advisable to check their availability through the University Library System at https://opac.uniba.it/easyweb/w8018/index.php? and contact the library to arrange the loan.		
Notes to the books		On the ADA e-learning website, the following resources are available: Lecture slides Examples of previous exams Examples of previous laboratory exams.		
Organization of the didactic activities				
Hours				
Total	Lectures	Practice sessions	Project work	Individual study
hours 225	56 hours	30 hours	0 hours	139 hours
CFU/ETCS				



9 CFU	7 CFU	2 CFU	0 CFU	
-------	-------	-------	-------	--

Teaching methods	
	Frontal lectures on the theoretical topics of the curriculum and practical examples to be developed in the classroom and at home. Laboratory activities to delve into practical and technological aspects. Periodic exercises (approximately weekly) to consolidate and deepen theoretical concepts. Furthermore, the extension of these exercises throughout the course duration serves as an incentive for attendance and self-assessment of the learning level. The exercises are designed for individual work.

Expected learning outcomes	
Knowledge and understanding	The student will acquire key knowledge regarding database models, query languages, and design methodologies commonly used at the conceptual and logical levels of database design. The student will possess formal knowledge enabling them to understand the structure of databases represented with the Entity-Relationship model (conceptual level), the relational model (logical level), and data queries expressed in the SQL language. Additionally, the student will learn to conceptually and logically design a database by analyzing collected requirements and applying specific methodologies to ensure desired properties in the implemented database. Furthermore, the student will develop the ability to efficiently query databases.
Applying knowledge and understanding	The student will acquire practical knowledge enabling them to apply their understanding of databases represented with the Entity-Relationship model (conceptual level), the relational model (logical level), and defined queries to create ER models, logical models, and SQL queries. Furthermore, the student will learn to practically implement a database in a specific DBMS and interface with it using Java.
Other skills	<i>Making judgements</i> The student will gain autonomy of judgment, as they will be able to decide on the strategies to apply during the conceptual and logical design phases of databases. In particular, they will be capable of evaluating aspects related to the completeness, correctness, and minimality of conceptual schemes, as well as efficiency aspects in designing logical schemes. <i>Communication</i> The student will be able to describe, through documents following precise methodologies, the choices made during the phases of conceptual and logical design. This will enhance their communication abilities towards potential users of the database. <i>Learning skills</i> The concepts learned will not only be useful for the topics covered during lectures but will also be applicable to different contexts (e.g., different languages or



	DBMS). This will enhance the student's ability to learn autonomously, allowing them to handle analogous situations, albeit different from those presented during the course.
--	--

Assessment	
Assessment methods	Evaluation is conducted through a written exam and a practical test. The written exam is structured with open-ended responses and serves as a prerequisite for the practical test, which takes place in the laboratory. Throughout the course, there are in-progress tests that are exemptive for the written exam. The score obtained in the written exam is retained for the entire academic year. The evaluation is on a scale of thirty for both exams, and the final grade is derived by averaging the scores of the assessments. Results are communicated to students through ADA for a preliminary view and subsequently on Esse3.
Evaluation criteria	The evaluation aims to assess the attainment of a strong problem-solving ability in the field of databases and information systems. Specifically, in relation to the expected learning outcomes, the following criteria are considered: Knowledge and Understanding · The student must demonstrate knowledge and understanding of the main conceptual and logical data models, database languages, database design methodologies, and architectural characteristics of database systems. These aspects are assessed in both exam tests. Applied Knowledge and Understanding · The student must demonstrate the ability to interpret informational needs in a given application context, formalize these needs through specifications, apply the mentioned models and methodologies for the realization of a database that meets the specifications, and query the created database. These aspects are evaluated in both exam tests. Autonomy of Judgment · The student must demonstrate the ability to formulate an independent critical judgment on different design solutions and support it within a working group. This autonomy of judgment is evaluated in the written exam. Communication Skills · The student must demonstrate the ability to communicate their design solution with language appropriateness, mastery of formal tools, and technical jargon. These communication skills are assessed in the written exam. Learning Ability · The student must demonstrate the ability to delve into the topics presented in lectures, tackling design problems of higher difficulty than those proposed during exercises. This learning ability is evaluated in both exam tests..
Measurements and final grade	Learning is assessed based on the ability to design and implement databases. The evaluation is founded on the mastery and completeness of the topics outlined in the curriculum.



Further information

It is recommended that students rely exclusively on information and communications provided on the official websites of the Department of Computer Science or on social groups only if they are established and managed solely by the instructors of the respective courses:

- <https://www.uniba.it/it/ricerca/dipartimenti/informatica/didattica/corsi-dilaurea/corsi-di-laurea>
- <https://www.uniba.it/it/ricerca/dipartimenti/informatica>
- <https://elearning.uniba.it/>

The course syllabi are available here:

- <https://elearning.uniba.it/course/index.php?categoryid=288>

The information that all students should know is detailed in the educational regulations and course handbooks available on the website:

- <https://www.uniba.it/it/ricerca/dipartimenti/informatica/didattica/corsi-dilaurea/corsi-di-laurea>

Students are advised to be cautious regarding information and materials circulating on unofficial websites or social groups, as they can often be unreliable, incorrect, or incomplete. For any doubts, students should arrange a meeting with the instructor following the specified office hours.

Link to the course on the department's ADA:
<https://elearning.uniba.it/course/view.php?id=2801>