

Professional English-2

Module 1

Lesson 1:

Workplace English: Present Simple vs Present Continuous

1) Present Simple (Routine, duties, facts)

When do we use it?

- **Routines / duties / responsibilities:** daily workflow, typical project tasks
- **General facts:** things that are always true

Workplace examples

- **I write 50 emails a day.**
- **She writes 50 emails a day.**

Rule (third person singular)

- With **he / she / it**, we usually add **-s**: **write = writes**
- Some verbs change differently:
 - **-es: fix : fixes** (verbs ending in **o, s, sh, ch, x, z**)
 - **-ies: copy : copies** (y → i + es)

Question form

- **Do / Does + subject + base verb**
- How many emails do you write every day?
- How many emails does Bill write every day?
Note: After “does”, the main verb does not take -s (wrong: *does writes*).

Negative form

- I don't write 50 emails a day.
- She doesn't write 50 emails a day.
In speaking, **don't / doesn't** are very common.

Common time expressions

- **always, usually, sometimes, rarely, never, every day/week/month, on Fridays**

2) Present Continuous (Now / temporary situations)

When do we use it?

- Actions happening right now
- Temporary actions happening around now (these days/this week)

Form

- **am/is/are + verb-ing**
- I'm checking my inbox now.
- This week I'm meeting clients from another country. (not necessarily right now, but this week)

Common time expressions

- **now, at the moment, currently, for the time being**

Bonus (repeated actions that feel annoying)

- **He is always sending me photos when I'm busy.**

3) Stative Verbs (usually not used with -ing)

Some verbs describe states, so they are not usually used in continuous forms:

- Feelings: **love, hate, like**
- Senses: **hear, see, feel**
- Mental activity: **believe, know, understand, mean, imagine**
Also common examples: **have** (possession), **want**

Quick test

Can you answer "What are you doing?" with that verb?

- **I am sleeping.** (action)
- **I am wanting.** (stative)

4) Quick choice guide (at work)

- "Usually / as a routine" → **Present Simple: I send reports every Friday.**
- "Now / temporary around now" → **Present Continuous: I'm preparing the release notes this week.**

Lesson 2 - Reading

A) Reading 1: “Guess the Job” (Gap Fill)

Text

1

A **Site Reliability Engineer (SRE)** works at the **intersection** of development and operations. Their **main goal** is to improve reliability and reduce **downtime**. They build automation, monitor systems, and respond to **incidents** when something breaks. They also help teams create a safer **release** process and improve the **deployment pipeline**. A good **Site Reliability Engineer (SRE)** communicates clearly during **outages** and helps prevent the same **issue** from happening again.

Question: What job is it?

B) “Guess the Job” (Gap Fill)

A **Test Engineer /QA Engineer** ensures that software works as expected before it reaches users. They design test cases, execute manual and automated tests, and report bugs with clear steps to reproduce. They also verify fixes, track test results, and collaborate with developers and product stakeholders to improve quality. A strong **Test Engineer /QA Engineer** thinks like a user, pays attention to details, and helps reduce risk in releases.

Question: What job is it?

Quick check (True/False)

1. This role mainly writes about new product features. (F)
2. This role creates test cases and reports bugs. (T)

Comprehension (short)

1. Name two responsibilities. **design test cases, execute manual and automated tests, and report bugs**
2. Who do they collaborate with? **developers and product stakeholders**

Match (Role vs. Responsibility)

Write **T** (Typical for Test Engineer) or **N** (Not typical).

1. Write clear steps to reproduce a bug (T)
2. Decide the company’s marketing strategy (N)
3. Create automated regression tests (T)
4. Approve a fix by re-testing it (T)
5. Design database schema from scratch (N)

C) Reading 3: “Skim and Find”

A **Product Security Engineer** helps teams ship software that is secure by design. They review code for common **vulnerabilities**, support **threat modeling**, and advise developers on safer **patterns**. They often work with **stakeholders** from engineering, **legal**, and **compliance** to make sure a product meets security requirements. When a new dependency is added, they may check its risk profile, licensing, and update policy. In many companies, this role also supports security training and builds **lightweight tools** to catch issues earlier in the pipeline.

2-minute task (Skimming):

Underline words related to:

- risk
- process
- teamwork

Comprehension (short):

1. Name two responsibilities.
2. Who do they collaborate with?

D) Reading 4: “Role Snapshot”

A Machine Learning Engineer (MLOps-focused) builds systems that make models usable in real products. They help move a model from a notebook to a **stable** service, track versions, and monitor performance in production. They work on data pipelines, testing, deployment, and rollback strategies. The job is not only about model accuracy; it is also about reliability, cost, and measurable impact.

Match (Role vs. Responsibility):

Write **M** (ML Engineer) or **N** (Not typical).

1. Set up model versioning and monitoring (**M**)
2. Design UI screens for mobile apps (**N**)
3. Build a data pipeline for training (**M**)
4. Manage rollback when a model fails (**M**)

Lesson 3 - Pronunciation

How do you pronounce these?

ROLES	
software engineer	product manager (PM)
backend developer	scrum master (agile role)
frontend developer	business analyst
full-stack developer	UX/UI designer
QA / test engineer	solution architect
DevOps engineer	data scientist
site reliability engineer (SRE)	machine learning engineer
security engineer	tech lead

Tech Terms			
Git	CI/CD	API	Docker
repository	pipeline	endpoint	container
branch	build	HTTP	Kubernetes
commit	test	status code	cluster
pull request (PR)	deploy	JSON	cloud
merge	rollback	token	server
code review	release	authentication	logging
issue	bug	authorization	monitoring
ticket	backlog	database	SQL
sprint	stand-up	query	schema

Lesson 4: Professional Email Writing in English

A practical guide to writing clear, formal, workplace emails

1) Core principles of a professional email

A good professional email helps the reader understand **what you want, why, and what happens next**—quickly.

- **Clarity first:** Make the purpose obvious in the first 1–2 sentences.
- **One email, one main goal:** Avoid mixing unrelated topics.
- **Reader-focused:** Explain what the recipient needs to do and by when.
- **Professional tone:** Polite, neutral, and direct (not emotional).
- **Scannable structure:** Short paragraphs, bullet points for details.
- **Concrete details:** Dates, times, versions, links, and owners.
- **Correctness:** Basic grammar and punctuation matter in formal contexts.

2) Standard structure (template)

- **Subject line** (what this email is about)
- **Greeting** (Dear/Hello + name/title)
- **Opening line** (context + reason for writing)
- **Purpose / request** (what you need)
- **Key details** (what, when, options, constraints)
- **Call to action** (next step + deadline if needed)
- **Closing line** (thanks + offer of help)
- **Sign-off** (Kind regards, Best regards, etc.)
- **Signature** (name, role, organization, contact)

3) Subject line patterns (with explanations)

A subject line should be short and informative. Use **keywords** that help the recipient search later.

Updates / information

- **Update on [Project/Topic]**
Use when you are sharing progress without asking for a decision.
- **Status update: [Project] (Week of [Date])**
Useful for recurring weekly reports.
- **Summary of [Meeting] on [Date]**
Use after a meeting to document decisions and actions.

Requests / actions

- **Request: [Action] by [Date]**
Best for clear, time-bound requests.
- **Approval needed: [Document/Change]**
Use when a formal approval is required.
- **Action required: [Task] (Deadline: [Date])**
Strong and direct—use when the item is urgent or blocking work.

Meetings

- **Meeting request: [Topic] (Proposed times)**
Use when you want to schedule and you offer options.
- **Agenda for [Meeting Name] on [Date]**
Use before a meeting to set expectations.

Issues / problems

- **Issue: [Problem] affecting [System/Users]**
Use for operational problems; include impact.
- **Incident report: [Service] on [Date]**
Use to document an outage/incident.

Documents

- **Document for review: [Title/Version]**
Use when you need feedback.
- **Submission: [Report Name] (vX.Y)**
Use when delivering a final or official version.

4) Greetings (and when to use them)

Your greeting sets the tone. More formal greetings are safer when you do not know the recipient well.

Very formal

- **Dear Mr. [Surname], / Dear Ms. [Surname],**
Use in formal business contexts, especially first contact.
- **Dear Dr. / Professor [Surname],**
Use in academic or professional contexts where titles matter.

Neutral and common at work

- **Dear [First Name],**
Professional and widely accepted.
- **Hello [First Name],**
Slightly less formal but still professional.

When you don't know the name

- **Dear Hiring Manager**, Good for job applications.
- **Dear Admissions Committee**, Good for universities.
- **Dear Support Team**, Good for customer/service emails.
- **To whom it may concern**, Very formal and less common now; use only when no better option exists.

For groups

- **Dear all**, Standard for internal group emails.
- **Hello everyone**, Friendly but still professional.

5) Opening lines (setting context) The opening line explains why you are writing and creates continuity.

Purpose-focused openings

- **I'm writing to follow up on [topic]**. Use when you want an update or reminder.
- **I'm reaching out regarding [topic]**. Neutral, useful for many situations.
- **I'm contacting you to request [something]**. Direct and clear.
- **I'm writing to share an update on [topic]**. Use for progress updates.

Reference previous contact

- **Following our meeting on [date], ...** Links your email to a shared event.
- **As discussed earlier today, ...** Confirms verbal agreements in writing.
- **Thank you for your email regarding [topic]**. Polite reply opener.

Polite "softener"

- **I hope you're doing well.** Neutral and common. Use once, not repeatedly.

6) Requesting politely (key phrases + meaning)

The goal is to be polite without sounding uncertain.

Direct but polite (recommended)

- **Could you please [verb] ...?** Standard polite request.
- **Would you be able to [verb] ...?** Slightly softer; good when asking for help.
- **I would appreciate it if you could [verb] ...** More formal; good for important requests.
- **Could you confirm whether ...?** Good for yes/no confirmation.
- **Please let me know if you can ...** Simple and professional.

Strong "action request"

- **Please review the attached [document] and share your feedback by [date].**
Clear action + deadline.
- **Could you approve [item] so we can proceed with [next step]?**
Shows why approval matters (reduces delays).
- **Please provide [information] at your earliest convenience.**
Formal, but avoid if you need a specific date—then include a deadline.
- **If possible, please share [item] by [date].**
Polite deadline request.

Avoid sounding too weak (common issue)

- **I was wondering if you could ...**
Very soft; can sound hesitant in professional settings. Use when you truly want to reduce pressure but prefer clearer forms above.

7) Giving information clearly (useful patterns)

- **For your reference, ...** Introduces background info.
- **Please note that ...** Highlights an important detail.
- **The key points are as follows:** Introduces bullet points.
- **Attached, you will find ...** Formal way to mention attachments.
- **Here is a brief summary:** Useful for busy recipients.

8) Asking for confirmation / decisions

- Could you please confirm by [date]?
- Please confirm whether you agree with the proposed approach.
- If you approve, we will proceed with [next step] on [date].
- Could you let me know your decision by [date]?

9) Deadlines and urgency (professional wording)

- **by [date]** (clear deadline)
- **at the latest by [date]** (firm but polite)
- **as soon as possible** (only if it's truly urgent; otherwise provide a date)
- **This is time-sensitive because [reason].** (adds justification)

10) Closing lines (polite endings)

Choose one based on the tone you want:

- **Thank you for your time and support.**
- **Thank you in advance for your help.** (use carefully; can feel pushy if overused)
- **Please let me know if you have any questions.**
- **I'm happy to provide more details if needed.**
- **Looking forward to your response.** (neutral)

11) Signoffs (formal to less formal)

- **Yours sincerely**, (very formal; use for official emails, especially to someone senior)
- **Sincerely**, (formal and safe for most official emails)
- **Kind regards**, (professional and very common; good default for student - instructor)
- **Best regards**, (slightly less formal; still professional and common)
- **Best**, (casual; mainly for internal/team emails, not for instructors unless you know them well)

12) Signature (what to include)

A professional signature typically includes:

- Full name
- Title / role
- Organization / department
- Phone (optional)
- Website / LinkedIn (optional, if relevant)

Good Practice

Topic: Request: Review of API Specification (v1.2) by 20 Feb

Dear Dr. [Surname],

I'm writing to request your review of the attached API Specification (v1.2) for the ColdLedger project. Could you please share your feedback by 20 February so we can proceed with implementation and finalize the endpoint list?

Key points to review:

- Endpoint naming and versioning strategy (v1 vs v1.2)
- Authentication/authorization approach (token scope and roles)
- Error responses and status code conventions
- Any missing requirements from the stakeholder perspective

If you prefer, I can also provide a short change log highlighting what changed since v1.1.

Thank you for your time and support.

Kind regards,

Elvan Duman

Topic: Meeting request: Release readiness review for Sprint 6 (30 minutes)

Hello [Name],

I'd like to schedule a short meeting to review release readiness for Sprint 6 and confirm the go/no-go decision for Friday's deployment. We will quickly align on open issues, test results, and rollback readiness.

Would any of the following times work for you (Turkey time)?

- Tue 18 Feb, 11:00–11:30
- Tue 18 Feb, 15:30–16:00
- Wed 19 Feb, 10:00–10:30

If none of these work, please suggest a suitable time.

Best regards,
Elvan Duman

Bad Practice

Topic: Homework

Hi,

I uploaded the homework. Please check it and give me my grade today.

Thanks.

Notes (Why this is NOT appropriate):

- **No identification:** The student does not write their full name, student ID, or class/section.
- **No course info:** The instructor cannot know which course (e.g., Software Engineering, Professional English, etc.).
- **No assignment details:** "Homework" is unclear (which homework? which topic? which week?).
- **No link/location:** The student does not say where it was uploaded (LMS, GitHub, email, etc.).
- **Unprofessional urgency:** "Give me my grade today" sounds demanding and unrealistic.

- **No proper closing/signature:** A formal email should end with a polite closing and signature.

Topic: Need help ASAP

Dear teacher,

My project does not work. I sent it before. Please fix it or tell me what to do. It is urgent.

Sent from my phone

Notes (Why this is NOT appropriate):

- **Greeting is too generic:** “Dear teacher” is not professional. Use “Dear Dr./Professor [Surname]” if possible.
- **No identification:** The student does not introduce themselves (name, ID, department, year).
- **No course/section:** The instructor cannot place the request in the correct course.
- **No technical details:** No error message, no steps to reproduce, no environment, no code link.
- **Vague reference:** “I sent it before” is unclear; the student should forward the original email or include a link.
- **Wrong request:** “Please fix it” is not appropriate for an instructor; the student should request guidance or feedback.
- **Urgency without a deadline/reason:** “ASAP/urgent” is not helpful; a specific deadline and context should be provided.

Topic: Homework 1 Submission – Access Confirmation Request – Ahmet Yilmaz
(Student ID: 221045067)

Dear Instructor,

My name is Ahmet Yilmaz (Student ID: 221045067). I am a second-year Software Engineering student enrolled in “Professional English for Software Engineering”.

I have submitted Homework 1 on the LMS. Could you please confirm that you can access my submission?

Submission title: Homework 1 – Tech Roles

Submission platform: LMS

Submission time: 16 February 2026, 21:10

Thank you for your time.

Kind regards,

Ahmet Yilmaz

Student ID: 221045067

Topic: Help Request: GitHub Actions CI/CD Pipeline Fails (npm ci exit code 1) – Ahmet Yilmaz (Student ID: 221045067)

Dear Instructor,

My name is Ahmet Yilmaz (Student ID: 221045067). I am a second-year Software Engineering student enrolled in “Professional English for Software Engineering”.

I am having an issue with my CI/CD pipeline on GitHub Actions. The workflow fails at the dependency installation step, and I am not sure how to interpret the error. Could you please advise me on what I should check first?

Project link: <https://github.com/ahmetyilmaz/se-project>

Environment: GitHub Actions (Ubuntu)

Error message: “npm ci failed with exit code 1”

Steps I followed:

1. I pushed my code to GitHub.
2. I opened the Actions tab and ran the workflow.
3. The pipeline failed during the install step.

My deadline is 18 February 2026. If possible, could you suggest a quick fix or recommend what I should read or practice to understand and resolve this type of issue?

Thank you very much.

Kind regards,

Ahmet Yilmaz

Student ID: 221045067