



Coding days

Tags: [Flow](#)

Coding days is the average number of days per week a developer contributes to the code base by making at least one commit.

In this article

[Which reports use Coding days?](#)

[What does Coding days measure?](#)

[How is Coding days calculated?](#)

[Example Coding days calculation](#)

[What data is included in Coding days?](#)

Which reports use Coding days?

Find Coding days in:

- [Check-in](#)
- [Trends](#)
- [Player Card](#)

[back to top](#)

What does Coding days measure?

A Coding day is any day where an engineer contributes code by making a commit. There are many different forms of engineering work: writing code, reviewing others' work, discussing specs and architecture. Arguably the most important of these is contributing code, and it's important for teams to make sure that time dedicated to things other than this stays at a reasonable level.

On large teams, it's difficult to see who is succeeding and who is struggling, or how the team as a whole is doing. Setting a standard around committing code every day is a simple yet powerful target to help ensure a project moves forward.

Do you know how many engineering days your team loses each week to meetings, planning, or other non-engineering tasks? If your team is like most, you're losing at least one day per week.

For example, say an engineering team has been asked to bring a third-party service in-house for financial reasons. Historically, this team has been a coding powerhouse and is seen as one of the most productive

engineering teams in the company.

Lately, however, something seems off. The engineers are grouchy and throughput is lower than normal.

The team lead looks at the team's Coding days trend and can see that the number has been slowly trending down over the last 12 weeks. From this information, the lead realizes the need for engineers to serve as de-facto product owners is causing a drop in their ability to commit code.

Knowing this, the team lead can jump in and run interference for the engineers who are affected most and have seen their Coding days fall off the most. This course correction helps bring the team's focus back to writing software.

Developers often have different amounts of Coding days and what's right for each team varies.

For example, developers working on ops or non-code work might only have one or two Coding days.

[back to top](#)

How is Coding days calculated?

Coding days is calculated as the total number of days where a developer commits code divided by the number of total weeks.

Total weeks is the total number of weeks in a period regardless of start date. It is the number of total days divided by 7.

Total days is the total number of days in the period minus days during weeks where no code was committed.

For organization and team calculations, Coding days is the sum of the coding days of each engineer in the period. Total days is the sum of the total days for each engineer in the period.

To find Coding days for partial weeks, where Total days is not divisible by seven, divide the number of Coding days by Total days. Then multiply by the lesser of seven and Total days.

$$\text{Coding days} = (\text{number of Coding days} / \text{Total days}) * \text{minimum}(7, \text{Total days})$$

[back to top](#)

Example Coding days calculation

Let's say we have two engineers on a team. One engineer codes three days on a particular week. The other engineer codes one day on that same week. The following week, the first engineer takes some much needed time off and therefore does not code. The second engineer codes five days.

Here's how Flow calculates the average coding days per week for that team:

1. Calculate total coding days. Flow sums the total coding days of each team member during the time period. Here the total coding days would be $3+1+5=9$.

2. Calculate total weeks. Flow then takes the sum of the total potential days each engineer could have committed code during that time period and divides it by 7. In this case, each engineer had 14 days. As a team of 2 engineers, it's $2*14=28$ days.
3. Calculate total days. In this case, one engineer took the entire second week off. The total days is therefore reduced to $28-7=21$ days, or three weeks.
4. Calculate average Coding days per week. Flow divides the total team coding days by the number of weeks found in step 3. Here, we get $9/3\cong 3$ for average Coding days per week.

[back to top](#)

What data is included in Coding days?

A day with a commit counts as a Coding day if the commit is:

- Authored by a user who isn't excluded from reports or a hidden user.
 - Authored by a user who is a contributor on a team when looking at team reports.
 - Not excluded by outlier detection.
 - Not a merge commit.
 - Not in a deleted branch or orphaned.
-

If you need help, please contact [Pluralsight Support](#).