



All Bootcamps

Hyperiondev

Debugging your code & *(your life for more effective...)* Time management

Lecture – Housekeeping

- ❑ The use of disrespectful language is prohibited in the questions, this is a supportive, learning environment for all - please engage accordingly.
- ❑ No question is daft or silly - **ask them!**
- ❑ There are Q/A sessions midway and at the end of the session, should you wish to ask any follow-up questions.
- ❑ For all non-academic questions, please submit a query:
www.hyperiondev.com/support
- ❑ Report a safeguarding incident:
<http://hyperiondev.com/safeguardreporting>
- ❑ We would love your feedback on lectures:
<https://hyperiondev.wufoo.com/forms/zsgv4m40ui4i0g/>

Github Repository – Lecture Examples/Slides

https://github.com/HyperionDevBootcamps/C4_SE_lecture_examples

Article

<https://www.coursera.org/articles/time-management-skills>

<https://www.mindtools.com/arb6j5a/what-is-time-management>

Objectives



1. Time management
 - a. Strategies
 - b. How to manage your time effectively
 - c. The benefits of effective TM
2. Debugging
 - a. Hypothesis-driven debugging
 - b. Debugging in VS code

Introduction

Time management is the process of organizing and allocating your time effectively to accomplish tasks, goals, and responsibilities. It involves consciously planning and prioritizing activities, setting goals, and utilizing techniques and tools to optimize productivity and efficiency.

Can you think of some benefits of effective time management?

- Increased productivity
- Reduced stress
- Improved work-life balance

Goal Setting

- Setting clear and achievable goals is essential in personal and professional life.
- Clear goals provide **direction**, **motivation**, and **focus**, helping you prioritize tasks and make informed decisions.
- They allow for **progress measurement**, **resource allocation**, and simplify **decision-making processes**.
- Clear goals also foster **accountability** and **personal growth**.

Goal Setting

The SMART (Specific, Measurable, Achievable, Relevant, Time-bound) goal-setting framework.



Goal Setting

Using the SMART goal-setting framework to reach a goal.

Example: Learning to code in 3 - 6 months

S : I will learn software engineering and become proficient in Python.

M: I will complete coding exercises and projects to demonstrate my understanding

A: I will dedicate consistent time each day to learning and practice, attend lectures, and actively engage in coding exercises and projects to reinforce my learning

R: Learning to code will enhance my career prospects as a software developer

T: I will achieve proficiency in software engineering, specifically Python, within the next 3-6 months."

Prioritization Techniques

- Eisenhower Matrix (urgent vs. important),



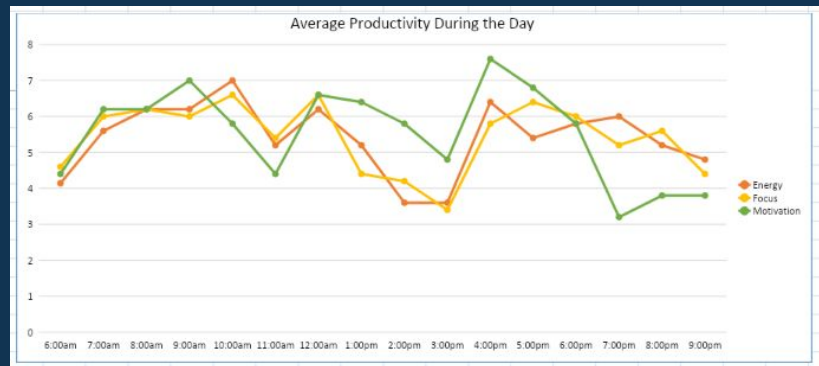
- ABCD Method (categorizing tasks),
- Pareto Principle (80/20 rule).

Focusing on high-value activities and minimizing time wasted on low-priority tasks.

Time Tracking and Analysis

- Use time tracking methods, such as using time management apps, spreadsheets, or journals.
- Analyze your time logs to identify patterns, time-wasting activities, and opportunities for improvement.

Week #1	6:00am			7:00am			8:00am			9:00am			10:00am			11:00am		
	Energy	Focus	Motivation	Energy	Focus	Motivation	Energy	Focus	Motivation	Energy	Focus	Motivation	Energy	Focus	Motivation	Energy	Focus	Motivation
Mon	5	5	3	6	8	9	6	7	8	9	8	10	8	6	8	5	4	6
Tue	6	6	7	4	5	6	5	5	6	3	5	4	5	8	4	6	7	3
Wed	3	5	3	6	8	9	6	7	8	9	8	10	8	6	8	5	4	6
Thu	6	5	6	8	6	4	9	10	6	5	5	8	8	8	4	5	8	3
Fri	2	2	3	4	3	3	5	2	3	5	4	3	6	5	5	5	4	4
Sat	4																	
Sun	3																	
Average	4	5	4	6	6	6	6	6	6	6	6	7	7	7	6	5	5	4



<https://blog.trello.com/find-productive-hours>

<https://clockify.me>

Effective Planning and Scheduling

Strategies for effective planning and scheduling, such as

- creating to-do lists
- using calendars or digital tools
- and blocking time for specific tasks

It is important that you set realistic deadlines and allow buffer time for unexpected interruptions.



Overcoming Procrastination

What are some common causes of procrastination?

- Perfectionism
- Overwhelm and task complexity
- Lack of motivation
- Fear of failure

Practical tips for overcoming procrastination, such as

- breaking tasks into smaller manageable parts
- setting deadlines
- utilizing rewards
- Considering the consequences

Managing Distractions

Common distractions in the workplace or daily life:

- Social media
- Email notifications
- Multitasking

Techniques to minimize distractions:

- turning off notifications
- creating dedicated focus time
- establishing boundaries



Time Blocking

A technique that involves dedicating specific blocks of time to specific tasks or activities.

Time blocking allows you to proactively plan and allocate time for different activities instead of relying on a to-do list or haphazardly tackling tasks as they arise. Here's how it works:

- **Identify Priorities:** Start by identifying your key priorities
- **Allocate Time Blocks:** Divide your day into blocks of time, typically in increments of 30 minutes to an hour. Assign each block to a specific activity or task.
- **Protect Time Blocks:** Treat your time blocks as sacred appointments.
- **Focus and Complete Tasks:** During each time block, focus solely on the assigned task or activity. Avoid multitasking and give your full attention to the current activity until it's completed or the time block ends.
- **Flexibility and Adaptation:** While time blocking provides structure, it's important to remain flexible and adapt as needed. Unforeseen circumstances or urgent matters may require adjustments to your schedule.

Batch Processing

Batch processing involves grouping similar tasks together and completing them consecutively, rather than addressing them individually throughout the day. Here's how batch processing works:

- **Task Categorization:** Identify tasks that can be grouped together based on their nature, requirements, or context. For example, responding to emails, making phone calls, or conducting research.
- **Dedicated Time Blocks:** Allocate specific time blocks for each task category.
- **Minimize Switching:** Instead of constantly switching between different types of tasks, focus solely on one task category during its assigned time block.

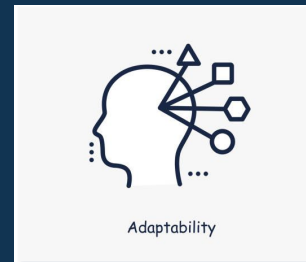
By focusing on similar tasks in batches, you reduce the time lost in task switching and the mental effort required to shift between different types of activities.

Multitasking can lead to as much as a 40% drop in productivity.

Multitaskers make up to 50% more errors when completing a task.

Continuous Improvement and Adaptation

- Continuously aim to improve in your time management practices.
- Regularly review and adjust your strategies based on your experiences and changing circumstances.
- While time blocking provides structure, it's important to remain flexible and adapt as needed. Unforeseen circumstances or urgent matters may require adjustments to your schedule.

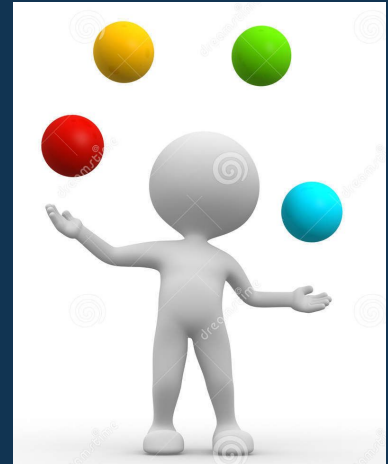


Stress Management

The ultimate goal of effective time management is to reduce stress.

Stress management techniques:

- Prioritizing self-care
- Taking breaks
- Maintaining a healthy work-life balance
- Avoid taking on too many responsibilities



Conclusion

Time management strategy:

- Goal setting
- Prioritization
- Time tracking
- Planning and scheduling
- Time blocking
- Batch processing
- Stress management

Long-term benefits of effective time management:

- Better time utilization
- Increased productivity
- Enhanced focus and concentration
- Improved decision-making
- Reduced stress
- Goal achievement
- Personal development
- Overall satisfaction and well-being

Hypothesis-Driven Debugging

Hypothesis-Driven Debugging

The key steps involved in this approach:

- Identify the problem or failure.
- Formulate a hypothesis about the cause.
- Design and conduct experiments to test the hypothesis.
- Analyze the results and refine the hypothesis if necessary.
- Repeat the process until the issue is resolved.

Benefits of Hypothesis-Driven Debugging

The advantages of using this approach in debugging:

- Provides a structured and systematic approach to problem-solving.
- Helps in narrowing down the potential causes of the issue.
- Saves time and effort by focusing on relevant experiments.
- Facilitates learning from debugging experiences and improving future debugging skills.

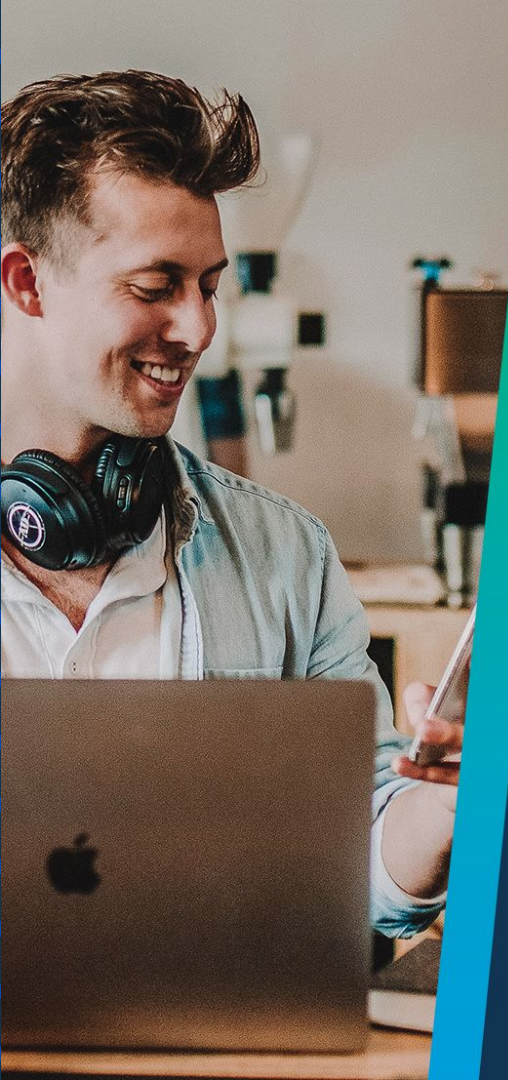
References

- <https://www.bing.com/images>
- <https://www.goskills.com/Office-Productivity/Resources/Time-batching>

Hyperiondev

Q & A Section

Please use this time to ask any questions relating to the topic explained, should you have any



Hyperiondev

Thank you for joining us

Stay hydrated
Avoid prolonged screen time
Take regular breaks
Have fun :)